

2

**NASA TECHNICAL  
MEMORANDUM**

NASA TM X-68181

NASA TM X-68181

(NASA-TM-X-68181) - A MESH GRADIENT  
TECHNIQUE FOR NUMERICAL OPTIMIZATION  
(NASA) 182 p HC \$11.25 CSCL 12A

X  
N73-17670

Unclas  
62785  
G3/19

**REPRODUCTION RESTRICTIONS OVERRIDDEN**

NASA Scientific and Technical Information Facility

**A MESH GRADIENT TECHNIQUE FOR  
NUMERICAL OPTIMIZATION**

by Edward Allen Willis, Jr.  
Lewis Research Center  
Cleveland, Ohio  
January 1973



*© override  
w/6-10-73*

Reproduced by  
**NATIONAL TECHNICAL  
INFORMATION SERVICE**  
US Department of Commerce  
Springfield, VA. 22151

COPYRIGHT BY  
EDWARD ALLEN WILLIS JR.

1973

Details of illustrations in  
this document may be better  
studied on microfiche

**REPRODUCTION RESTRICTIONS OVERRIDDEN**

NASA Scientific and Technical Information Facility

A MESH GRADIENT TECHNIQUE FOR  
NUMERICAL OPTIMIZATION

ABSTRACT

by

EDWARD ALLEN WILLIS, JR.

This paper deals with a class of successive-improvement optimization methods in which directions of descent are defined in the state space along each trial trajectory. The given problem is first decomposed into two discrete levels by imposing mesh points. Level I then consists of running optimal subarcs between each successive pair of mesh points. For normal systems, these optimal two-point boundary value problems can be solved by following a routine prescription if the mesh spacing is sufficiently close. A spacing criterion is given. Under appropriate conditions, the criterion value depends only on the coordinates of the mesh points, and its gradient with respect to those coordinates may be defined by interpreting the adjoint variables as partial derivatives of the criterion value function. In Level II, the gradient data is used to generate improvement steps or search directions in the state space which satisfy the boundary values and constraints of the given problem. The family of feasible varied trajectories thus constructed converges to the "nearest" locally-optimum trajectory, if any such exist.

## ACKNOWLEDGEMENTS

The graduate study program culminating in this dissertation has been partially supported by the National Aeronautics and Space Administration, under provisions of the Federal Employees' Training Act. It is a real pleasure to acknowledge not only this formal support, but also the encouragement and moral support given by my colleagues and other friends at the Lewis Research Center. Special mention is due to Dr. Bernard Lubarsky and Messrs. Milton A. Beheim, Hugh H. Henneberry, and Richard J. Weber.

Thanks are also due to the Chairman and Members of the examining committee for their thoughtful and constructive review. The Chairman, Professor Eli Reshotko, deserves credit "beyond the call of duty" for his guidance and steadfast support of this project. I am also grateful to acknowledge the many interesting discussions, constructive criticism and useful suggestions contributed by Dr. Gilmer Blankenship. Credit is also due to Drs. I. Greber, L. S. Lasdon, D. Macko, S. K. Mitter, J. D. Pearson, and J. D. Schoeffler for valuable, stimulating discussions.

Last but not least, I wish to thank my wife Jean for showing the "patience of a saint" during difficult times.

# TABLE OF CONTENTS

<u>Chapter/Section</u>	<u>Page</u>
1. SUMMARY . . . . .	1
2. INTRODUCTION . . . . .	3
2.1 Problem Considered . . . . .	6
2.2 Existing Methods of Solution . . . . .	7
2.3 The Mesh-Gradient (MG) Approach. . . . .	11
2.4 Comments . . . . .	12
3. THE MESH-GRADIENT TECHNIQUE. . . . .	15
3.1 Level I - Subarc Solutions . . . . .	16
3.1.1 The Transition Matrix Algorithm. . . . .	16
3.1.2 Comments . . . . .	21
3.1.3 Conditions for Feasible Solutions. . . . .	23
3.1.4 Example - Zermelo's Problem. . . . .	29
3.2 Level II - Iteration in Mesh Point Space, $X_{pp}$ . . . . .	32
3.2.1 Interpretation of the Adjoint Variables as $^{pp}$ Partial Derivatives. . . . .	32
3.2.2 Necessary Conditions in $X_{pp}$ . . . . .	38
3.2.3 Direction-of-Descent Algorithms. . . . .	42
4. SAMPLE APPLICATIONS. . . . .	48
4.1 Example 1 - Minimum Effort Control . . . . .	48
4.1.1 Descent Via Programming. . . . .	49
4.1.2 Descent by First or Second-Order Steps . . . . .	52
4.2 Example 2 - A Problem with Conjugate Points . . . . .	53
4.3 Example 3 - Control of an Unstable Van De Pol Oscillator . . . . .	59
4.3.1 Computing Codes. . . . .	60
4.3.1.1 Descent by Modified Fletcher-Powell Program. . . . .	61
4.3.1.2 Descent by Second-Order Steps. . . . .	61
4.3.1.3 Generating Initial Guesses for Lower Level Iteration. . . . .	61
4.3.2 Numerical Results. . . . .	64
4.3.2.1 Comparison With Previous Methods . . . . .	66
4.3.2.2 Constrained Problems . . . . .	72
4.3.2.3 Effect of Poor Starting Iterates . . . . .	74
4.3.2.4 Effect of Varying the Distribution of Computational Effort . . . . .	75
4.4 Example 4 - Optimal Space Trajectories . . . . .	78
4.4.1 Subarc Solutions . . . . .	81
4.4.1.1 Boundary Conditions. . . . .	82
4.4.1.2 General-Case and Low Thrust Solutions in Terms of the Fundamental Matrix . . . . .	83
4.4.1.3 Solutions for the Case of Impulsive Thrust . . . . .	86
4.4.2 Mesh-Point Iteration . . . . .	86

<u>Chapter/Section</u>	<u>Page</u>
4.4.3 Numerical Results. . . . .	90
4.4.3.1 Accuracy of the Approximate Solution . . . . .	90
4.4.3.2 Low Variable Thrust Solutions. . . . .	95
4.4.3.3 Impulsive Thrust Solutions . . . . .	97
5. CONCLUDING REMARKS . . . . .	104
6. BIBLIOGRAPHY . . . . .	108
7. MAIN SYMBOLS . . . . .	125
8. APPENDIX A: DISCUSSION OF LEVEL I - THE TWO POINT BOUNDARY VALUE PROBLEM . . . . .	129
8.1 First-Order Necessary Conditions . . . . .	129
8.1.1 Basic Definitions and Hypotheses . . . . .	133
8.1.2 The Maximum Principle for the Basic Optimal Control Problem. . . . .	133
8.1.3 Extensions to More General Optimal Control Problems . . . . .	135
8.2 Second-Order Necessary Conditions and Sufficient Conditions for a Local Minimum . . . . .	137
8.2.1 Convexity of Legendre-Clebsch Condition. . . . .	137
8.2.2 Normality Condition. . . . .	137
8.2.3 The Jacobi or No-Conjugate-Point Condition . . . . .	138
8.2.4 Equivalence of Second-Order Conditions . . . . .	138
8.3 Convergence of the Transition Matrix Algorithm . . . . .	141
9. APPENDIX B: DISCUSSION OF LEVEL II - SUCCESSIVE IMPROVEMENT STEPS IN STATE SPACE . . . . .	147
9.1 Kuhn-Tucker Necessary Conditions . . . . .	147
9.2 Descent Via First-Order Gradient Steps . . . . .	148
9.2.1 Auxiliary Variational Problem. . . . .	149
9.2.2 Linear Step Algorithm. . . . .	152
9.2.3 Ultimate Convergence Rates . . . . .	152
9.3 The Hessian Matrix and Descent Via Second-Order Steps. . . . .	155
9.3.1 The Hessian. . . . .	156
9.3.2 Quadratic Step Algorithm . . . . .	158
9.3.3 Convergence. . . . .	160
9.4 Descent Via Fletcher-Powell Algorithm. . . . .	
10. APPENDIX C: AN ANALYTICAL APPROXIMATION TO THE FUNDAMENTAL MATRIX FOR SPACE TRAJECTORIES. . . . .	162

## LIST OF FIGURES

<u>Figure</u>	<u>Title</u>	<u>Page</u>
2-1	Trajectory with Intermediate Constraints. . . . .	5
3-1	Initial and Final Stages in the Lower Level Cycle . .	18
3-2	Zermelo's Problem . . . . .	30
3-3	Geometry of the Attainable Sets . . . . .	37
4-1	Initial-Guess Data for Problem 4.1. . . . .	50
4-2	Solution Curves for Problem 4.2 . . . . .	55,56
4-3	Directions of Descent for Problem 4.2 . . . . .	58
4-4	Program Organization, Fletcher-Powell Method. . . . .	62
4-5	Program Organization, Second-Order Method . . . . .	63
4-6	Comparison of Techniques. . . . .	67,68
4-7	Effect of Terminal and Intermediate Boundary Conditions. . . . .	73
4-8	Effect of Varying the Distribution of Computational Effort Between Levels . . . . .	77
4-9	Typical Multi-Burn Space Trajectory . . . . .	79
4-10	Boundary Conditions for High - Low Thrust Subarc . . .	79
4-11	Schuler Frequencies Along a 270 <sup>0</sup> , 250-Day Earth-to- Mars Rendezvous Trajectory. . . . .	91
4-12	Criterion Value Vs. Iteration No. . . . .	93
4-13	Effect of Number of Mesh Points on Criterion Value Obtained. . . . .	94
4-14	Comparison of Direct and Indirect Trajectories. . . .	96

<u>Figure</u>	<u>Title</u>	<u>Page</u>
4-15	Comparison of Two- and Multiple-Impulse Trajectories. . . . .	99
4-16	Characteristics of Optimal 3-Impulse Earth-to-Neptune Rendezvous Trajectories . . . . .	102
8-1	Schematic of Transfer Matrix Algorithm. . . . .	142

#### LIST OF TABLES

<u>Table</u>	<u>Title</u>	<u>Page</u>
4-1	Conjugate Gradient Step Data. . . . .	51
4-2	No. of Steps Required for 0.5% Convergence. . . . .	69
4-3	Estimated Run Times for Different Methods . . . . .	71
4-4	Iteration History from Poor Initial Trajectory. . . . .	75

## 1. SUMMARY

This paper deals with a class of successive-improvement optimization methods in which directions of descent are defined in the state space along each trial trajectory. The given problem is first decomposed into two discrete levels by imposing mesh points. Level I then consists of running optimal subarcs between each successive pair of mesh points. For normal systems, these optimal two-point boundary value problems can be solved by following a routine prescription if the mesh spacing is sufficiently close. A spacing criterion is given. Under appropriate conditions, the criterion value depends only on the coordinates of the mesh points, and its gradient with respect to those coordinates may be defined by interpreting the adjoint variables as partial derivatives of the criterion value function. In Level II, the gradient data is used to generate improvement steps or search directions in the state space which satisfy the boundary values and constraints of the given problem. The family of feasible varied trajectories thus constructed converges to the "nearest" locally-optimum trajectory, if any such exist.

This approach reduces the typical deterministic optimal control problem to an exercise in the classical theory of maxima and minima. It also leads to a class of apparently novel algorithms for optimal

trajectory computations. These differ as to their ultimate rates of convergence and computational sophistication, but all share the same basic operation - namely, decomposition and variation of the state trajectory itself. This derives maximum benefit from the initially-given constraint and boundary value information. It is also useful for solving computationally-unstable problems (where the integration interval is long compared to state or adjoint system time constants), by taking the mesh-spacing small enough. Therefore, the present class of methods prove relatively effective when applied to unstable problems or problems with numerous boundary values and constraints, as illustrated by several examples.

## 2. INTRODUCTION

Many important physical processes are describable by systems of deterministic ordinary differential equations. Since there are often more variables than equations, it makes sense to use some of the extra degrees of freedom as control variables - in order to optimize a criterion of merit and to meet the boundary values prescribed for the process. Thus (with many technicalities deferred until the formal "statement of the problem" in Section 2.1) the systems studied may typically have the form

$$\text{minimize } J = \int_{t_0}^{t_f} f_0(\vec{x}(t), \vec{u}(t), t) dt$$

with

$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{u}(t), t)$$

subject to various boundary conditions and constraints on the state vector  $\vec{x}(t)$  and the control vector  $\vec{u}(t)$ .

From a mathematical viewpoint, modern theories, such as, L. S. Pontryagin's Maximum Principle (ref. 137; references are listed alphabetically in Chapter 6, Bibliography) give enough "necessary conditions" to define solutions to most practical problems. Unfortunately, it is not always easy to implement these conditions in

practical numerical studies. Experience indicates that the difficulties experienced in conducting numerical studies are usually due to one or both of the following causes:

(a) Numerical instability results, when the integration interval exceeds the dominant state or adjoint time constant, in computer overflows, inordinate sensitivity of final condition to small perturbation of the initial data, etc. That this is a widespread problem may be inferred by recalling that for a simple linear system, each stable state equation will give rise to an unstable adjoint equation, and conversely.

(b) Numerous constraints on the state trajectory (c.f. fig. 2-1), involving point or path, equality or inequality relations, normally entail the enforcement of additional relations, such as the "jump" and "corner" conditions or appropriate forms of the transversality conditions. Typically, one must introduce and determine an extra set of auxiliary multipliers for every intermediate constraint, thus adding to the computational burden. Philosophically, one would expect the presence of intermediate constraints to help, not hinder the optimization process by reducing the range and dimension of the space to be searched.

These factors, which are greatly compounded by the nonlinearity and high dimensionality typical of practical problems, are responsible for phenomena such as numerical instability and multiple or nonexistent solutions which seriously hinder the effective conduct

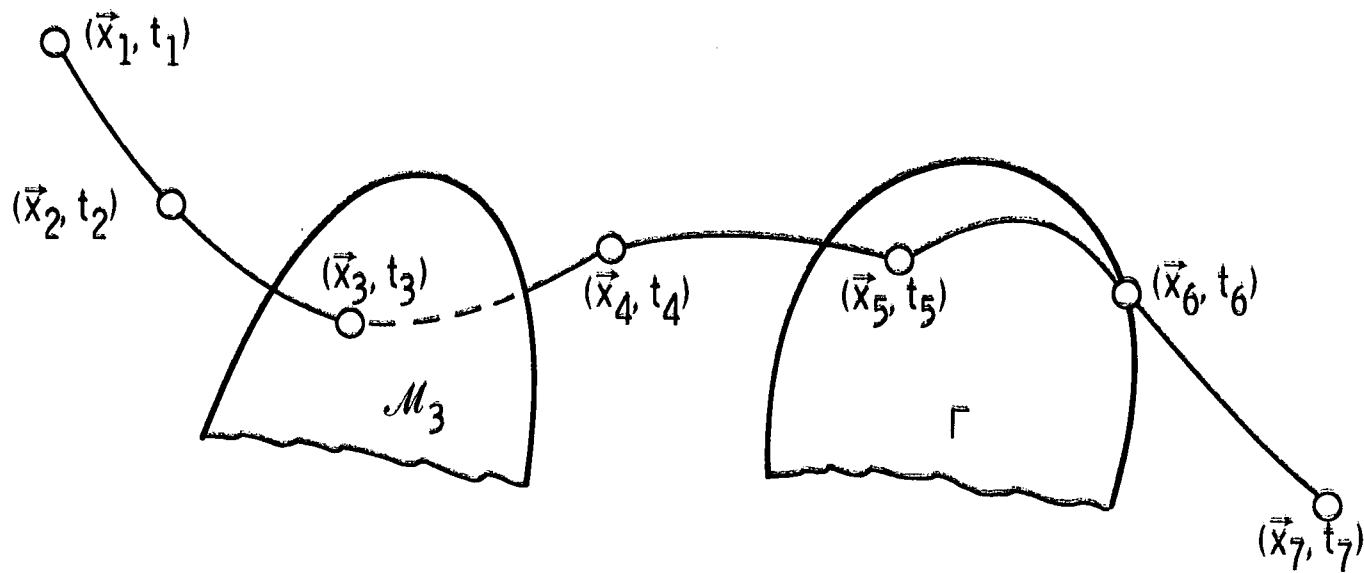


FIGURE 2-1. - TRAJECTORY WITH INTERMEDIATE CONSTRAINTS.

of numerical studies. With this comment as background, the general problem to be considered here may be stated in the following form.

### 2.1 Problem Considered

Minimize the following criterion function over  $\vec{u}$  and  $\vec{\beta}$ :

$$J = \int_{t_0}^{t_f} f_0(\vec{x}(t), \vec{u}(t), \vec{\beta}, t) dt + \text{P.V.C.} \quad (1)$$

where the abbreviation P.V.C. stands for point-value contributions (to be defined later), the state variables

$$\vec{x}(t) \in R_N \quad \text{for all } t \in \mathcal{T} = [t_0, t_f]$$

are defined by

$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{u}(t), \vec{\beta}, t) \quad (2)$$

the admissible controls are defined by

$$\vec{u}(t) \in \Omega \subset R_M \quad \text{for all } t \in \mathcal{T} \quad (3)$$

and  $\vec{\beta} \in R_L$  is a vector of constant (but adjustable) design parameters.

In addition there are terminal- and intermediate-point equality constraints, i.e.,

$$\vec{x}(t_k) \triangleq \vec{x}_k \in \mathcal{M}_k(\vec{x}; t) \quad (4)$$

where  $t_0 < t_1 \dots t_K = t_f$  and the  $\mathcal{M}_k$  are  $n_k$  dimensional smooth manifolds ( $0 \leq n_k \leq N$ ).

Finally, state-variable inequality constraints may also apply, i.e.,

$$\vec{x}(t) \notin \text{int } \Gamma \quad (5)$$

where  $\Gamma \subset R_N$  represents one or more excluded regions of state space.

## 2.2 Existing Methods of Solution

Necessary conditions and sufficient conditions for basic forms of the problem are well in hand, as may be judged from the appearance recently of comprehensive texts such as Athans & Falb (ref. 4), Bryson & Ho (ref. 24), and Lee & Markus (ref. 108). Several basic computational approaches are presently available for producing numerical solutions in specific cases. Since recent surveys (e.g., refs. 3, 54, 96, 132 and 149) display literally hundreds of individual contributions, the present discussion will be limited to broad categories, with details left to the indicated references.\*

Conceptually, the most powerful method now available is Dynamic Programming. Although a valuable theoretical tool, its computational application have been very limited due to extremely high storage requirements. Hence, despite the appearance of D.P. algorithms with decreased storage requirements (refs. 97 and 98), it is felt that iterative successive-improvement are more promising as a class.

First-order schemes such as Kelley's control iteration (ref. 81) are perhaps useful for generating initial guesses for higher ordered schemes. Unfortunately, they converge slowly in the neighborhood of the solution and tend to produce inaccurate adjoint trajectories.

---

\*References are listed and categorized in Chapter 6, Bibliography.

These disadvantages, which are pointed out and discussed in connection with Figures 5 and 6 of reference 100, render first-order techniques generally unsuitable for the present needs, and they will not be further discussed.

Newton-type methods achieve quadratic convergence in the neighborhood of the true solution, by making use of linear system theory. The Transition Matrix approach (c.f. ref. 24) is one example of a Newtonian (second order) boundary iteration technique. The major disadvantage of this method is its tendency toward numerical instability. As previously explained either the state or adjoint equations are necessarily unstable. This can result in the state and adjoint variables being of entirely different orders of magnitude (e.g.,  $e^{at}$  vs  $e^{-at}$ ) - which implies the transition matrix is ill-conditioned - when the time interval significantly exceeds the system's dominant time constant.

To reduce numerical-instability problems, the "backward-sweep" approach can be applied to both boundary-iteration (ref. 24) and control-iteration (ref. 123) techniques. As applied to boundary iteration, this entails integrating the state, adjoint and Ricatti matrix equations backwards from assumed terminal conditions. Terminal values are then adjusted by iteration so that the desired initial conditions are achieved. The advantage of this approach is that the adjoint equations are being integrated in the "stable" direction. The Ricatti equation may also be tested for system "normality" and the presence of conjugate points (c.f. Section 8.2,

Appendix). On the other hand, even though the stability problem may be somewhat eased in some cases, it is by no means eliminated. This is because the canonical equations necessarily contain both stable and unstable components and these are merely interchanged by reversing the sense of integration.

When control rather than boundary-value iteration is used, the state equations are integrated forward using an assumed control history and the resulting trajectory stored; the adjoint and Ricatti matrix equation are then integrated backwards to determine an improved control history which is both more nearly optimal and more nearly feasible. Stability is greatly improved because the state and adjoint equations are decoupled during any one pass. Nevertheless, a fairly good initial guess is still required for the entire control history, and (short of first making use of a lower-order scheme) there is no obvious or systematic way to obtain one. In addition, this technique requires both forward and backward integration passes with storage of the entire state and adjoint trajectories and Ricatti matrix histories. It requires fixed-step-size integration schemes which, in general, are less efficient than variable step size schemes. Finally, the process does not inherently meet all boundary conditions and an additional multiplier must be determined for each "hard" terminal constraint (i.e., one which must be satisfied exactly). On balance, this approach is rather attractive for free-terminal problems but rapidly loses its appeal when "hard" constraints are added.

Finally, the best known of the Newtonian functional approximation techniques is that of Quasilinearization (ref. 14). The procedure is to simply linearize the canonical equations around trial state/adjoint trajectories. The following linear perturbation equations

$$\begin{bmatrix} \delta \dot{\vec{x}} \\ \delta \dot{\vec{\psi}} \end{bmatrix}_i = \begin{bmatrix} \vec{f}(\vec{x}_i, \vec{\psi}_i) - \dot{\vec{x}}_i \\ \vec{g}(\vec{x}_i, \vec{\psi}_i) - \dot{\vec{\psi}}_i \end{bmatrix} + \begin{bmatrix} \frac{\partial (\vec{f}, \vec{g})}{\partial (\vec{x}, \vec{\psi})} \end{bmatrix}_i \begin{bmatrix} \delta \vec{x}_i \\ \delta \vec{\psi}_i \end{bmatrix} \quad (7)$$

$$\vec{x}_{i+1}(t) = \vec{x}_i(t) + \delta \vec{x}_i(t); \quad \vec{\psi}_{i+1}(t) = \vec{\psi}_i(t) + \delta \vec{\psi}_i(t) \quad (8)$$

are solved iteratively until a suitable norm, e. g.,  $\max_{t \in \mathcal{J}} \left| \delta \vec{x}_i(t) \right|$ , becomes "sufficiently" small. The procedure unfortunately is subject to instability whenever the integration interval significantly exceeds the dominant time constant and also is dependent on having a good initial guess for both the state and adjoint trajectories. The technique also requires storage of the entire state-adjoint trajectories and is not adaptable to efficient, variable-step-size numerical integration routines.

From the preceding discussion it may be concluded that none of the existing methods seems to be specifically addressed to the most general, unstable, and highly constrained form of the problem. Moreover, when nonlinear effects are added to the previously-mentioned instability and constraint problems (c.f. Fig. 2-1), it is readily understandable why computational efficiency is still a major consideration despite the capabilities of modern computer facilities.

State variable inequality constraints or intermediate-equality constraints are particularly troublesome in that they entail additional mathematical conditions and may require a laborious "patching together" of free and constrained arcs. Philosophically, it would seem that the additional a priori information represented by the state constraints should be usable to help, not hinder, in computing the solution.

### 2.3 The Mesh-Gradient (MG) Approach

Based on the preceding remarks, the present approach was designed with two primary objectives in view: (a) to take maximum advantage of the a priori data given about the problem in the form of intermediate state-variable constraints; and (b) to eliminate or avoid the problem of numerical instability.

In essence, the present approach relies upon the principle of decomposition to divide an intractable given problem into nested levels of individually-manageable subproblems. Specifically the procedure is to impose a set of mesh points  $\{\vec{x}_k; t_k\}$  leading from  $(\vec{x}_0; t_0)$  to  $(\vec{x}_K; t_K)$ , certain members of which belong to the constraint manifolds  $\mathcal{M}_j$ . This resolves the problem into two levels.

Level I: Select an initial mesh,  $\{(\vec{x}_0; t_0), (\vec{x}_1; t_1) \dots (\vec{x}_K; t_K)\}$  and (with the mesh-points held fixed) solve the resulting sequence of Two-Point Boundary Value Problems (TPBVP) using any suitable numerical optimization technique. These solutions may be readily

computed by taking closely-spaced mesh points; this avoids the stability problems often associated with initial-value methods, and results in an initial feasible trajectory satisfying the given physical boundary values and constraints.

Level II: Optimize the mesh-point locations so as to minimize  $J$ , using any suitable mathematical programming technique. Note that here the physical constraints and boundary values merely reduce the range and/or dimensionality of the search.

#### 2.4 Comments

(a) If a certain controllability condition is satisfied, numerical instabilities may always be avoided in Level I by taking a sufficiently close mesh spacing. This is an important point, because Level I is nested inside Level II and must operate through many cycles for every step in the Level II search. Thus, it is essential for the present method to have a reliable and efficient method of solving short subarc TPBVP's.

(b) If the TPBVP solutions are optimal, then  $J$  is a function of the mesh-point coordinates only. The gradient of  $J$  may be readily defined in terms of the adjoint-variable and Hamiltonian-function discontinuities at each mesh point. Furthermore, the Hessian matrix may be derived from the subarc transition matrices. Thus, Level II consists of a conventional, well-posed mathematical programming problem for which a very satisfactory and complete theory is available - c.f. Fiacco & McCormick (ref. 45).

(c) The selection of an initial set of mesh points is to some extent a matter of judgment. As a minimum, the initial and final points, plus one point for every junction with a constraint manifold, must be included. Additional unconstrained points may be inserted to satisfy the mesh-spacing criterion for stability (to be developed later).

(d) The tractability of closely spaced TPBVP's depends on a controllability assumption - namely, that each mesh point is in fact reachable from the preceding one. If this is not true in general (as is apt to be the case with fixed-time, bounded-control problems), then care must be used to avoid the appearance of abnormal (i.e., unfeasible) subarcs. In many cases, the TPBVP's may be reformulated in terms of quantities that are attainable with the available control. This would typically involve dropping redundant or uncontrollable coordinates, substituting iteration parameters, etc. See reference 115 for instance. Otherwise, penalty functions may be used to weigh temporarily-unavoidable boundary value or control-constraint violations.

(e) The Level I and Level II operations individually make use of established techniques and theories. The novelty and contribution of the Mesh-Gradient approach consists in combining these techniques in such a way as to avoid numerical instabilities and to make effective use of the constraint data specified for the problem. Whereas, prior techniques may be broadly classified as involving boundary

iteration, control iteration or successive approximation (i.e., quasilinearization), the characteristic feature of the present approach is iteration in the state space.

(f) This dissertation is organized in the following manner. The basic Mesh-Gradient Technique is presented in Chapter 3 (with all but the most essential of the supporting material deferred to Chapters 8 and 9). Four computational examples are treated in detail in Chapter 4, while conclusions and recommendations for further study are given in Chapter 5. References are listed and categorized in Chapter 6, main symbols are presented in Chapter 7, and the remaining Chapters are Appendices.

### 3. THE MESH-GRADIENT TECHNIQUE

The major steps comprising the "Mesh-Gradient" method are presented in this chapter. For the sake of clarity, only its characteristic features and underlying assumptions are discussed here, with many derivations and other lengthy details deferred until Chapters 8 and 9.

As applied to Problem 2.1, the technique begins by introducing a set of mesh points  $\{x_k; t_k\}$ , which in some sense lead from the given initial conditions to the desired final conditions. The points are ordered, so that  $t_0 < t_1 < \dots < t_{k-1} < t_k < \dots < t_K = t_f$ , and initially arranged to lie along the analyst's best guess at the optimal trajectory. The collection of admissible mesh points in  $R_{NK}$  is denoted by  $X_{pp}$ ; other main symbols are listed in Chapter 7. One member of  $X_{pp}$  belongs to each constraint manifold, and additional, unconstrained points may be introduced for computational convenience. This defines a sequence of two point boundary value problems (TPBVP) connecting adjacent pairs of mesh points. The objective of Level I is that each TPBVP solution will be optimal. When this holds true, it follows that the criterion value  $J$  is a function of the mesh point coordinates and times only, and its gradient with respect to

these coordinates is well-defined. The notation  $\nabla_{\mathbf{x}}( )$  will denote the gradient with respect to feasible direction in  $\mathbf{x}_{pp}$ . Thus, the objective of Level II is to achieve that  $J$  is minimal with respect to feasible variations in  $\mathbf{x}_{pp}$ , i.e., that  $\nabla_{\mathbf{x}} J = 0$  and that  $\nabla_{\mathbf{x}}[\nabla_{\mathbf{x}} J]$  is positive definite.

### 3.1 Level I - Subarc Solutions

As mentioned in Chapter 2 it is crucial for the present technique to have an efficient, reliable, hopefully almost fool-proof method of solving TPBVP's. This is because Level I is nested inside Level II and must operate through a full cycle (i.e., solve  $K$  TPBVP's) for every "function-evaluation" in the Level II search. In this section we will discuss the chosen technique, and subsequently the major conditions and restrictions that apply to it.

For smooth, well-behaved functions of practical engineering interest (c.f., the standard definitions and assumptions listed in Section 8.1.1) it is well known that an optimal control and its response must satisfy the Maximum Principle (c.f., Sections 8.1.2 and 8.1.3), an appropriate form of the transversality condition (Section 3.2.1) and also the convexity or strengthened Legendre-Clebsch condition, the normality condition and the Jacobi or No-Conjugate-Point condition (c.f., Sections 8.2.1-8.2.3). From henceforth it is assumed that these conditions apply to individual subarcs.

#### 3.1.1 The Transition Matrix Algorithm

While many numerical techniques are available for solving two point boundary value problems, Newtonian iteration based upon linear

perturbation theory (the transition matrix algorithm or TMA) is used here because of its efficiency and because the associated second-partial matrices can also be used in the upper level calculations. Specifically, after introducing the usual adjoint variables ( $\vec{\psi}(t)$ ) and Hamiltonian ( $\mathcal{H}$ ) and using the maximum principle to eliminate the control variables,\* the subarc problem may be expressed in the canonical form

$$\left. \begin{aligned} \dot{\vec{x}}(t) &= \frac{\partial \mathcal{H}}{\partial \vec{\psi}} = \vec{f}(\vec{x}(t), \vec{\psi}(t), \vec{\beta}, t) \\ \dot{\vec{\psi}}(t) &= - \frac{\partial \mathcal{H}}{\partial \vec{x}} = \vec{g}(\vec{x}(t), \vec{\psi}(t), \vec{\beta}, t) \\ \vec{x}(t_k) \triangleq \vec{x}_k &\in \mathcal{M}_k \quad \text{and} \quad \vec{x}(t_{k+1}) \triangleq \vec{x}_{k+1} \in \mathcal{M}_{k+1} \end{aligned} \right\} \quad (1)$$

where the  $\vec{x}_k \in \mathcal{X}_{pp}$ , with  $k = 1, 2, \dots, K$ .

With assumed initial values of  $\vec{\psi}(t_k)$ , Eqs. (1) may be integrated forward to obtain end points  $\vec{x}_*(t_{k+1})$  (see Fig. 3-1) which in general differ from the desired points  $\vec{x}_{k+1}$ , and contributions  $J_k$  to the criterion value. Relative to the given initial points  $\vec{x}_k$  and the actually obtained final points  $\vec{x}_*(t_{k+1})$ , each  $J_k$  is at least stationary in  $F_{sa}$ . Moreover, it will be shown that  $J_k$  depends upon its end-point coordinates only, and that (with suitable normalization) the partial derivatives of the integral contribution to  $J_k$  with respect to  $\vec{x}(t_k)$  and  $\vec{x}_*(t_{k+1})$  are given by

\* If  $\Omega$  is unbounded, it is generally possible (under the hypotheses of Section 8.1.1) to solve for  $\vec{u}$  as an explicit, differentiable function of  $\vec{x}, \vec{\psi}, \vec{\beta}$  and  $t$ . If  $\Omega$  is bounded, however,  $\vec{u}$  may depend implicitly on some components of  $\vec{\psi}$ , i.e., via the location of switching boundaries. In that case, some elements of the Jacobian matrices in Eqs. (6) and (7) below will be represented by impulse functions, and special techniques must be used to integrate the transition matrix Eqs. (4).

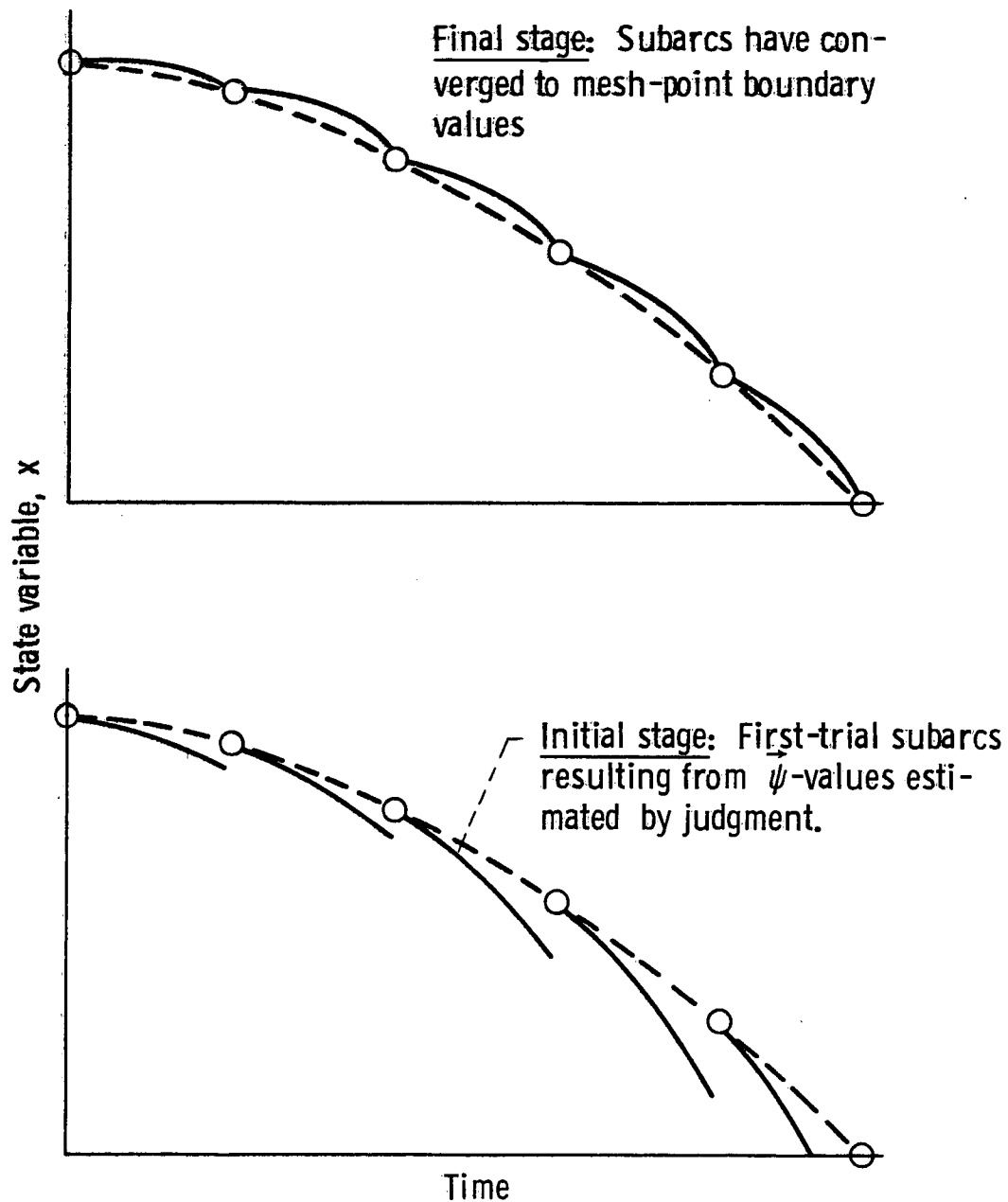
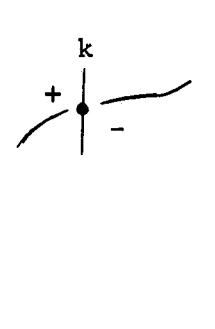


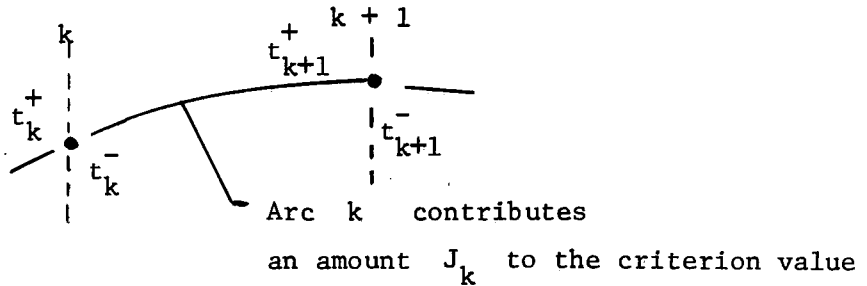
FIGURE 3-1. - INITIAL AND FINAL STAGES IN THE LOWER LEVEL CYCLE.

and

$$\left. \begin{aligned} \frac{\partial J_k}{\partial \vec{x}(t_k)} &= -\vec{\psi}(t_k^-) \\ \frac{\partial J_k}{\partial \vec{x}_*(t_{k+1})} &= \vec{\psi}(t_{k+1}^+) \end{aligned} \right\} \quad (2)$$


where superscripts  $+$  and  $-$  denote left and right hand limits, respectively. When P.V.C. terms are present in  $J$ , Eq. (2) must be extended to include  $\frac{\partial}{\partial \vec{x}(t_k)} (\text{PVC})_k$  and  $\frac{\partial}{\partial \vec{x}_*(t_{k+1})} (\text{PVC})_{k+1}$ , respectively.

The required values of  $\vec{\psi}(t_k)$ , which cause  $\vec{x}_*(t_{k+1})$  to coincide with the desired point  $\vec{x}_{k+1}$ , are found by linearizing Eqs. (1) around the trajectory ensuing from a previous ( $j^{\text{th}}$ ) estimate  $\vec{\psi}_j(t_k)$ . In the customary way this yields the transition matrix and perturbation system.



For the present purposes it will be convenient to partition the transition matrix  $\Phi(t_0; t)$  into  $4 N \times N$  blocks, i.e.,

$$\Phi(t_0; t) = \begin{bmatrix} A & B \\ C & D \end{bmatrix}_{t_0; t} \quad (3)$$

With

$$\frac{d}{dt} \begin{bmatrix} \overline{A} & \overline{B} \\ \underline{C} & \underline{D} \end{bmatrix}_{t_0;t} = \begin{bmatrix} \overline{\alpha} & \overline{\beta} \\ \underline{\gamma} & \underline{\delta} \end{bmatrix}_t \begin{bmatrix} \overline{A} & \overline{B} \\ \underline{C} & \underline{D} \end{bmatrix}_{t_0;t} \quad (4)$$

and

$$\begin{bmatrix} \overline{A} & \overline{B} \\ \underline{C} & \underline{D} \end{bmatrix}_{t_0;t_0} = I \text{ (the } 2N \times 2N \text{ identity matrix)} \quad (5)$$

where  $\alpha$ ,  $\beta$ ,  $\gamma$ , and  $\delta$  are  $N \times N$  Jacobian matrices formed by differentiating  $\vec{f}$  and  $\vec{g}$  with respect to  $\vec{x}$  and  $\vec{\psi}$  in turn, namely

$$\alpha = \frac{\partial \vec{f}}{\partial \vec{x}}, \quad \beta = \frac{\partial \vec{f}}{\partial \vec{\psi}}, \quad \gamma = \frac{\partial \vec{g}}{\partial \vec{x}}, \quad \delta = \frac{\partial \vec{g}}{\partial \vec{\psi}} \quad (6)$$

Recalling the canonical definitions of  $\vec{f}$  and  $\vec{g}$  (c.f., Section 8.1.2), the Jacobians may be expressed (component-wise) as

$$\left. \begin{aligned} \alpha_{ij} &= \frac{\partial^2 \mathcal{H}}{\partial x_i \partial \psi_j} & \beta_{ij} &= \frac{\partial^2 \mathcal{H}}{\partial \psi_i \partial \psi_j} \\ \gamma_{ij} &= - \frac{\partial^2 \mathcal{H}}{\partial x_i \partial x_j} & \delta_{ij} &= - \frac{\partial^2 \mathcal{H}}{\partial \psi_i \partial x_j} \end{aligned} \right\} \quad (7)$$

Because of the continuity and differentiability of the original function  $\vec{f}(\vec{x}, \vec{u}, \vec{\beta}, t)$ , it may be observed that  $\beta$  and  $\gamma$  are symmetric, i.e.,

$$\left. \begin{aligned} \beta &= \beta^T & \gamma &= \gamma^T \\ \delta &= - \alpha^T \end{aligned} \right\} \quad (8)$$

Thus, the Jacobians entail  $2N^2 + N$  independent quantities rather than  $4N^2$ .

Then, computing  $\Phi(t_0; t)$  as usual and adopting the notation that  $A(t_k; t_{k+1}) = A_k$ , etc., the solution of the perturbation system may be written as

$$\begin{bmatrix} \vec{\delta x}(t_{k+1}) \\ \vec{\delta \psi}(t_{k+1}) \end{bmatrix}_{\text{iteration } j} = \begin{bmatrix} A_k & B_k \\ C_k & D_k \end{bmatrix}_j \begin{bmatrix} \vec{\delta x}(t_k) \\ \vec{\delta \psi}(t_k) \end{bmatrix}_j \quad (9)$$

Finally, assuming that  $B_k^{-1}$  exists, we may solve for  $\vec{\delta \psi}(t_k)$  with  $\vec{\delta x}(t_k) = 0$ , yielding an increment

$$\vec{\delta \psi}_j(t_k) = B_k^{-1} \vec{\delta x}_j(t_{k+1}) \quad (10)$$

which (for linear dynamics) would zero the predicted terminal error  $\vec{\delta x}_{j+1}(t_{k+1})$ .

### 3.1.2 Comments

(1) It is shown in Section 8.3 that the sequence

$$\begin{aligned} \vec{\psi}_0(t_k) & \quad \text{arbitrary} \\ \vec{\psi}_{j+1}(t_k) &= \vec{\psi}_j(t_k) + \vec{\delta \psi}_j(t_k) \end{aligned} \quad (11)$$

converges either quadratically or not at all to the solution of (1).

(2) Moreover, the assumptions of Section 8.1.1 imply that except in the presence of an abnormal or conjugate point trajectory, convergence can always be obtained if adjacent partition points are properly arranged and sufficiently close together (briefly, this means that each point is to be attainable from the preceding one and that the interval  $[t_k; t_{k+1}]$  does not much exceed the dominant local time-constant of the canonical system);

(3) The results of section 3.2.1 below imply that, under the normalization implied in Eq. (2), the blocks of the transition matrix

$\Phi = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$  may be interpreted as

$$\begin{aligned} A_k(t) &= \frac{\partial \vec{x}(t_{k+1})}{\partial \vec{x}(t_k)} \\ B_k(t) &= \frac{\partial \vec{x}(t_{k+1})}{\partial \vec{\psi}(t_k)} \\ C_k &= \frac{\partial \vec{\psi}(t_{k+1})}{\partial \vec{x}(t_k)} \\ D_k &= \frac{\partial \vec{\psi}(t_{k+1})}{\partial \vec{\psi}(t_k)} \end{aligned} \tag{12}$$

for all  $t \in \mathcal{T}$ .

(4) The existence and proper interpretation of the partial derivatives indicated in this section and below depend upon (a) the continuity, differentiability and convexity properties listed in Section 8.1.1, and (b) the non-singularity of  $B_k$ . The latter does not necessarily follow from the former, however, and the conditions under which  $B_k^{-1}$  or its equivalent exists are further discussed in the next section.

To summarize, the Transition Matrix Algorithm or TMA comprises the following steps:

- (a)  $\vec{\psi}_0(t_k)$  arbitrary
- (b) solve Eqs. (1) to (9)

(c) Compute the linear increment  $\delta \vec{\psi}_j(t_k)$  and correct  $\vec{\psi}_j(t_k)$  per Eqs. (10) and (11).

(d) Iterate steps (a)-(c) until a suitable error norm, say

$$\delta \vec{x}_j^T(t_{k+1}) Q \delta \vec{x}_j(t_{k+1}) \quad (12)$$

(where  $Q$  is a positive definite matrix) becomes sufficiently small.

For a TPBVP, this scheme requires the forward integration of the  $2N$  cononical equation plus the 4 sets of  $N \times N$  matrix equations to define  $A$ ,  $B$ ,  $C$  and  $D$ . It may be recalled from Eqs. (7) that  $\begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}$  is symmetric so that only  $2N^2 + N$  terms must be computed. There is no need to store the trajectory histories. At the final time the  $N \times N$  matrix  $B$  must be inverted, and used to compute the derived matrices  $E$ ,  $F$ ,  $G$  and  $H$  which will be defined in Section 3.2. As shown in the Appendix (8.2.4) these matrices collectively contain only  $2N^2 + N$  independent components.  $E$ ,  $F$ ,  $G$ , and  $H$  may be stored in the same locations used for  $A$ ,  $B$ ,  $C$ , and  $D$ , and used prior to the next iteration to compute quantities that are of interest for Level II.

### 3.1.3 Conditions For Feasible Solutions

As has been previously implied, even the generous hypotheses of Appendix A are not sufficient to guarantee the existence of a matrix  $B_k^{-1}$  which solves Eq. (1) in 3.1.1. This is because, even though Problem 2.1 is assumed to be well-posed, i.e., its solution exists in principle, the system is in general only partially controllable

in the sense of Section 8.1.1. In such a case we have no way of knowing apriori whether one mesh point lies in the controllable subspace of the next, etc. Thus, the concept of and conditions for feasibility are of central importance here.

A feasible trajectory is one satisfying all specifications of the problem except that the criterion  $J$  is not necessarily minimized. That is, using the attainable set notation defined in Section 8.1.1, a trajectory  $\vec{x}(t)$ , joining the points  $\vec{x}(t_k) = \vec{x}_k$  and  $\vec{x}(t_{k+1}) = \vec{x}_{k+1}$ , does not exist unless

$$\vec{x}_{k+1} \in \mathcal{A}(\vec{x}_k, t_k; t_{k+1})$$

or equivalently,

$$\vec{x}_k \in \mathcal{R}(t_k; t_{k+1}, \vec{x}_{k+1}) \quad (1)$$

An analogous statement with respect to sets of traversal holds between every pair of distinct points belonging to a feasible trajectory, i.e., the relation

$$\vec{x}(t) \in \mathcal{A}^*(\vec{x}_k, t_k; t) \cap \mathcal{R}^*(t; t_{k+1}, \vec{x}_{k+1}) \quad (2)$$

holds, for any  $t \in (t_k, t_{k+1})$ .

Local feasibility hypothesis. - The feasibility condition of Eq. (2) is obviously satisfied by any pair of points belonging to an optimal trajectory. The condition also applies, by continuity, in some neighborhood of the optimal trajectory. The local feasibility hypothesis consists in assuming that condition (2) applies along an arbitrary trial trajectory. This implies that all of the necessary

2-point transfers exist. If the hypothesis is untrue for the original problem (as evidenced by the appearance of singular B), it can be relaxed into a controllable version of the problem by temporarily introducing point-discontinuity penalty functions. That is, the "hard" boundary conditions such as  $\vec{x}(t_{k+1}) = \vec{x}_k$  are removed, and in their places a P.V.C. such as the quadratic form

$$p_j [\vec{x}(t_{k+1}) - \vec{x}_k]^T Q [\vec{x}(t_{k+1}) - \vec{x}_k] \quad (3)$$

(where Q is a positive definite N x N matrix and  $p_j > 0$  is a scalar penalty factor) is added to the previous criterion value.

Then as shown in reference 45, the sequence  $p_0$  arbitrary  $> 0$ ,  $p_j > p_{j-1}$ , with  $\lim_{j \rightarrow \infty} p_j = \infty$ , will lead to a trajectory with vanishingly small mesh point discontinuities, from which the solution to the original problem may be recovered.

In many cases, however, it is possible to avoid the use of penalty functions by approximately choosing the iteration parameters. Also, the adjustable mesh-points should be specified in a manner compatible with the initial boundary conditions. For example, for time-open or time-optimal problems, the mesh points should not be fixed in time. That is, one must formulate the TPBVP's in a fashion that makes sense in terms of the given system and the capabilities of the control system, by describing the mesh points in terms of quantities that are attainable for this particular system. Unfortunately, this process does require some judgement on the part of the analyst and cannot be reduced to a routine prescription. The

following considerations, however, are helpful in identifying the proper formulation.

Generalized iteration parameters. - As has been mentioned the two point boundary value problems are solved numerically by iteration. In general, the  $N$  terminal conditions are to be met by proper choice of  $N$  initial conditions. Let the vector  $\vec{d}_k$  abstractly symbolize the desired terminal condition and  $\vec{p}_{k-1}$  the initial condition to be determined.

Typically, the components of  $\vec{d}(t_k)$  are determined by  $n_k$  terminal equality constraints and  $N - n_k$  transversality conditions. For example, if  $\mathcal{M}_k$  requires the first  $n_k$  components of  $x(t_k)$  to have fixed final values, the other  $N - n_k$  being free, then  $\vec{d}(t_k)$  has the form

$$\vec{d}(t_k) = \left[ x_1(t_k), x_2(t_k) \dots x_{n_k}(t_k); \psi_{n_k+1}(t_k) \dots \psi_N(t_k) \right]^T \quad (4)$$

and must satisfy the terminal condition

$$\vec{d}(t_k) = \left[ x_k|_1, x_k|_2 \dots x_k|_{n_k}; 0 \dots 0 \right]^T \quad (5)$$

Similarly,  $\vec{p}_{k-1}$  represents the initial value of those components of the state and adjoint vector which are not defined by boundary values or transversality conditions at time  $t_{k-1}$ . That is,  $\vec{p}_{k-1}$  represents  $N$  independent initial conditions which lead, via integration of the canonical equations, to  $N$  dependent final values  $\vec{d}(t_k)$ . Thus, the iteration problem is to find the  $N$ -vector  $\vec{p}_{k-1}$  which will transfer the canonical state-adjoint system to a terminal state that satisfies Eqs. (4) and (5). For such a transfer to

exist, it is necessary that the symbolic,  $N \times N$  Jacobian matrix

$$\frac{\partial \vec{d}(t_k)}{\partial \vec{p}_{k-1}} \quad (6)$$

be nonsingular. This requirement dictates the choice of the iteration parameters  $\vec{p}_{k-1}$ .

It is desirable, whenever possible, to use the initial adjoint variables as the  $\vec{p}_{k-1}$ 's; however, other parameters can be used when appropriate to avoid abnormal or unfeasible subarcs. For example, the elapsed or final time may be included if the control effort available is bounded. This is because an arbitrary mesh point  $\vec{x}_k$ , even if it satisfies the local feasibility hypothesis of Eq. (2) with respect to the preceding and subsequent mesh points, then may not belong to  $\mathcal{A}^*(\vec{x}_{k-1}, t_{k-1}; t_k)$  for all values of  $t_k$ .

For example, even the elementary problem

$$\text{minimize } J = \frac{1}{2} \int_0^t |u(t)|^2 dt$$

with

$$\dot{x}(t) = u \quad x(0) = 0 \quad x(1) = 1 + \epsilon$$

(where  $\epsilon > 0$ ) exhibits this behavior if the control magnitude is constrained, such that  $|u(t)| \leq 1$ . No solution then exists unless the final time is relaxed at least to  $1 + \epsilon$ .

In more general examples, it may be possible to use components of the design parameter vector  $\vec{\beta}$  or of the state or control vectors, in addition to the final time, to replace some components of

$\vec{\psi}(t_0)$ . See reference 115 for a relatively thorough discussion of this possibility.

Finally, note that it is possible to investigate the existence of  $B^{-1}$ , at least over a short interval, by examining the transition-matrix differential Eq. (4) in 3.1.1. I.e., to the first order in time,

$$B(t - t_0) \approx [\alpha(t_0)B(t_0) + \beta(t_0)D(t_0)](t - t_0) = \beta(t_0)(t - t_0) \quad (7)$$

Hence, if the Jacobian matrix  $\beta(t_0)$  itself is singular, one need not solve Eqs. 3.1.1(1) through 3.1.1(10) in order to learn that  $B$  is singular. Usually, an examination of the rank and structure of  $\beta(t_0)$  will suggest the choice of generalized parameters  $\vec{p}_{k-1}$ .

Mesh spacing. - The convergence properties of the TMA are developed in Section 8.3. Given the present hypothesis that the generalized Jacobian (6) is not singular, it is shown that quadratic convergence is obtainable if either the initial guess  $\vec{\psi}_0(t_k)$  is "close enough"\* or the time interval is short enough.\*\* "Short enough" means that the time interval does not greatly exceed the dominant local "time constant" of the system. The latter can perhaps be judged from the physics of the system, or may be derived from the Jacobian matrices 3.1.1(7), evaluated at particular times of interest.

More practically, a heuristic scheme could be employed to decrease mesh spacing whenever convergence difficulties are noted -

\*C.f. Eqs. 8.3(7) and 8.3(8).

\*\*C.f. Eqs. 8.3(10) through (12).

as evidenced, for example, by an excessive number of trials to reach convergence.

#### 3.1.4 Example-Zermelo's Problem

The following example will illustrate the transition matrix algorithm. It also demonstrates the importance of choosing proper iteration parameters as discussed in Section 8.1.1. As Figure 3-2 indicates, a boat has a constant speed  $v$  relative to the water, which in turn is moving with a fixed velocity  $u$  in the X-direction. It is required to find the heading angle  $\alpha$  to minimize the time of transit between two fixed points. The system is

$$\dot{x} = u + v \cos \alpha \quad \dot{y} = v \sin \alpha \quad (1)$$

Upon applying the maximum principle and other steps prescribed above, it is readily seen that the optimal control law is

$$\tan \alpha = \psi_2 / \psi_1 \quad (2)$$

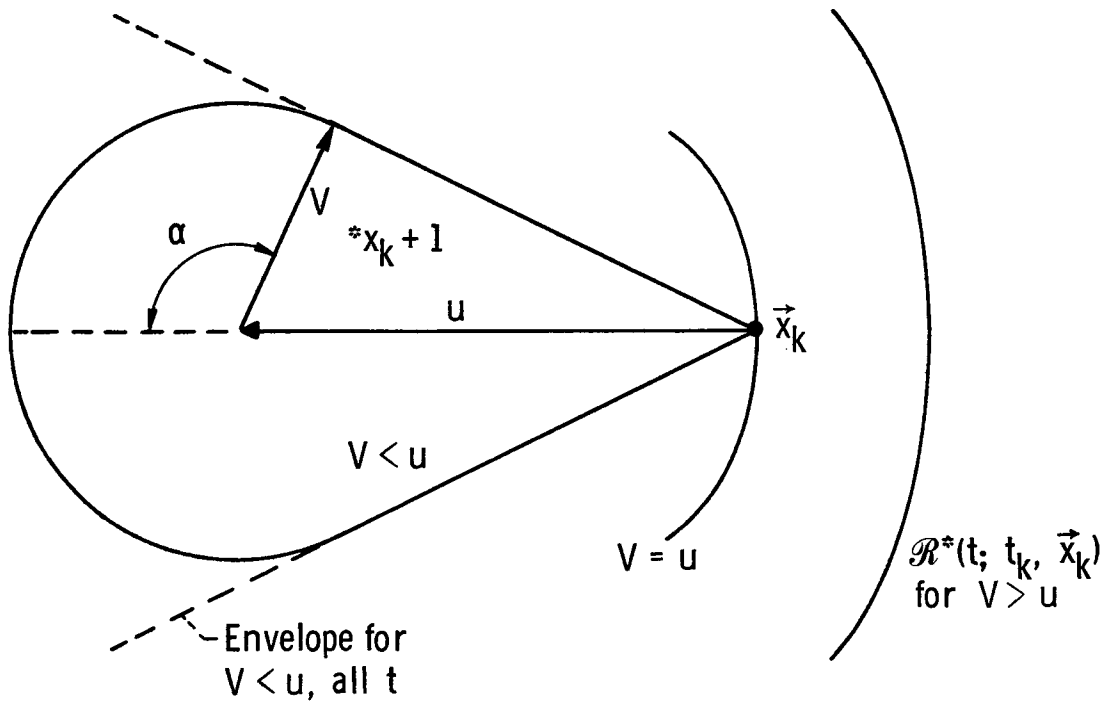
and the canonical equations are

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi}_1 \\ \dot{\psi}_2 \end{bmatrix} = \begin{bmatrix} u + v\psi_1/p \\ v\psi_2/p \\ 0 \\ 0 \end{bmatrix} \quad (3)$$

where

$$p = \left[ \psi_1^2 + \psi_2^2 \right]^{1/2} \quad (4)$$

Unfortunately, if  $\psi_1$  and  $\psi_2$  are taken as iteration parameters, it can be shown that the  $B$  matrix has a determinant proportional to



REVERSED TRAVERSAL SETS FOR ZERMELO'S PROBLEM.

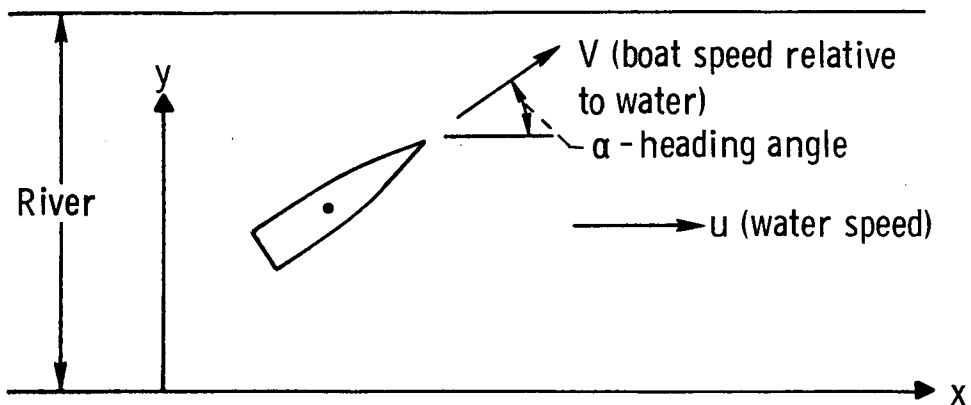


FIGURE 3-2. - ZERMELO'S PROBLEM.

$$\begin{vmatrix} -\psi_1^2/p + 1 & -\psi_1\psi_2/p \\ -\psi_1\psi_2/p & -\psi_2^2/p + 1 \end{vmatrix} \equiv 0 \quad (5)$$

to the first order in time, so that  $B$  is singular and hence the pair  $(\psi_1, \psi_2)$  are not proper iteration parameters. On the other hand, the pair  $(\alpha, t_f)$  results in a tractable result, i.e., to the first order

$$B = \frac{\partial(x, y)}{\partial(\alpha, t_f)} = \begin{pmatrix} -v \sin \alpha & u + v \cos \alpha \\ v \cos \alpha & v \sin \alpha \end{pmatrix} \quad (6)$$

whose determinant

$$\Delta B = -v^2 + uv \cos \alpha \quad (7)$$

does not vanish if  $v > u$ . But, if  $v \leq u$ ,  $B$  is singular at headings such that

$$\alpha = \cos^{-1}(v/u) \quad (8)$$

Figure 3-3 illustrates the geometry of the traversal sets  $\mathcal{R}^*(t_1; t_k, x_k)$  near the point  $(x_k, t_k)$ , corresponding to  $v > u$  (the oval),  $v = u$  (the circle) and  $v < u$  (the pointed figure).

In the latter case, singular  $B$  would be experienced if a trial mesh point  $\vec{x}_{k-1}$  lying outside the envelope were chosen. In that case a penalty function formulation as discussed above could be used to recover a feasible mesh point.

### 3.2 Level II - Iteration in Mesh Point Space, $X_{pp}$

At this point the selected mesh-points have been connected by individually-optimal subarcs, and the machinery exists to re-connect them after perturbations. It remains only to perturb the points in

$\mathbf{X}_{pp}$  so that the overall transfer is optimal also. Clearly, if we can show that: (a) the contribution to  $J$  arising from each subarc is a function of its mesh coordinates only; and (b) define the gradient of  $J$  in  $\mathbf{X}_{pp}$ , then all the existing theories of mathematical programming may be incorporated in Level II. We demonstrate these two crucial points below, before proceeding with the development search algorithms in  $\mathbf{X}_{pp}$ .

### 3.2.1 Interpretation of the Adjoint Variables as Partial Derivatives

Theorem. - Consider Problem I, the optimal TPBVP defined on the interval  $\mathcal{T} = [t_0, t_1]$ , with  $\vec{f} = \vec{f}(\vec{x}, \vec{u})$  only and no P.V.C. terms.\*

Assume that:

(a) The state variable derivatives  $\vec{f}(\vec{x}, \vec{u})$  are continuous and has continuous second derivatives in all arguments and the component  $f_0(\vec{x}, \vec{u})$  is bounded below for all  $t \in \mathcal{T}$ ;

(b) The admissible controls consist of all bounded, piecewise-continuous functions, with finite number and magnitude of discontinuities, whose values  $\vec{u}(t)$  belong to a convex set  $\Omega \subset \mathbb{R}_M$  for all  $t \in \mathcal{T}$ ;

(c) The two mesh points  $\vec{x}_0$  and  $\vec{x}_1$  and times  $t_0$  and  $t_1$  defining the terminals of the trajectory are chosen so that discontinuities in  $\vec{u}(t)$ , if present, will coincide with  $t_0$  and/or  $t_1$ . Thus, it is sufficient here to consider only single, continuous functions with values in  $\Omega$ ; and

---

\*This entails no loss of generality since the P.V.C. terms are already of the desired form and the neglected parameters  $\vec{\beta}$  and  $t$  may be regarded as additional state variables - c.f., 8.1.3.

(d) The control  $\vec{u}(t)$  defined over  $\mathcal{I}$  and its corresponding trajectory  $\vec{x}_u(t)$  are unique, feasible, and optimal. That is,  $\vec{x}(t_0) = \vec{x}_0$ ,  $\vec{x}(t_1) = \vec{x}_1$  and the Maximum Principle and the Convexity, Normality and Jacobi conditions are satisfied for all  $t \in \mathcal{I}$ .

Then. - (a) the resulting integral criterion value depends upon  $\vec{x}_0$  and  $\vec{x}_1$  only, i.e.,  $J = J(\vec{x}_0, \vec{x}_1)$  in  $\mathcal{F} \subset R_{2N}$ . ( $\mathcal{F}$  is the domain of feasible pairs of initial and final points in  $R_{2N}$ , for given  $\mathcal{I}$ ); and (b) the gradient of  $J(\vec{x}_0, \vec{x}_1)$  exists and is defined in  $R_{2N}$  by

$$\nabla J(\vec{x}_0, \vec{x}_1) = \begin{bmatrix} -\vec{\psi}(t_0) \\ +\vec{\psi}(t_f) \end{bmatrix} \quad (1)$$

within a scalar multiple.

Proof. - For condition (a), optimality implies that an admissible control exists in the form  $\vec{u}_{\text{opt}}(t) = \vec{v}(\vec{x}_0, \vec{x}_1, t)$ , for all  $t \in \mathcal{I}$ , which solves problem I. Thus, (a) follows immediately upon inserting this function into the system equations and the defining integral for  $J$ .

The existence of expression (1) is also implied by optimality. That is, in the rigorous proof of the Maximum Principle (c.f., Chapters 1 and 2 of ref. 137) it is shown that for  $\vec{u}(t)$  and  $\vec{x}_k(t)$  to be optimal, it is necessary that there exist a non-zero, continuous vector function  $\vec{\psi}(t) = (\psi_0(t), \psi_1(t), \dots, \psi_N(t))^T$  and a Hamiltonian function  $\mathcal{H}(\vec{\psi}(t), \vec{x}(t), \vec{u}(t)) = \vec{\psi}(t)^T \vec{f}(\vec{x}, \vec{u})$  such that

$$\dot{\vec{\psi}} = \frac{-\partial \mathcal{H}}{\partial \vec{x}} \quad (2)$$

and the Hamiltonian is maximized with respect to  $\vec{u} \in \Omega$  for all  $t \in \mathcal{I}$ .

Thus, it remains to show that the adjoint variables may also be interpreted as partial derivatives of the integral criterion value  $J$ . This may be readily accomplished, under the present assumptions, by simply constructing the first variation,  $\delta J$ . Since  $\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{u}(t))$  in problem I, we may write

$$J = \int_{t_0}^{t_1} \left[ f_0(\vec{x}(t), \vec{u}(t)) + \vec{\psi}^T(t) \{ \dot{\vec{x}}(t) - \vec{f}(\vec{x}(t), \vec{u}(t)) \} \right] dt$$

Recalling the definition of  $\mathcal{K}$  and setting  $\psi_0 = -1$ , this may be expressed as

$$J = \int_{t_0}^{t_1} [-\mathcal{K} + \vec{\psi}^T(t) \dot{\vec{x}}(t)] dt \quad (3)$$

Integrating the last term by parts we obtain

$$J = \vec{\psi}^T(t_1) \vec{x}(t_1) - \vec{\psi}^T(t_0) \vec{x}(t_0) - \int_{t_0}^{t_1} [\mathcal{K} + \dot{\vec{\psi}}^T(t) \vec{x}(t)] dt \quad (4)$$

The variation  $\delta J$  due to perturbations in  $\vec{x}(t)$  and  $\vec{u}(t)$  may now be calculated by differentiating under the integral sign in (4):

$$\begin{aligned} \delta J = & \vec{\psi}^T(t_1) \delta \vec{x}(t_1) - \vec{\psi}^T(t_0) \delta \vec{x}(t_0) \\ & - \int_{t_0}^{t_1} \left[ \left( \frac{\partial \mathcal{K}}{\partial \vec{x}} + \dot{\vec{\psi}} \right)^T \delta \vec{x} - \frac{\partial \mathcal{K}}{\partial \vec{u}} \delta \vec{u} \right] dt \end{aligned} \quad (5)$$

(This is justified since  $\vec{f}$ , and hence  $\mathcal{K}$ , have continuous second derivatives. Note that if any discontinuities in  $\vec{u}(t)$  are present, they occur at mesh points only and do not affect the preceding results.)

The first term in the integral vanishes identically because of Eq. (2). And the second term vanishes if the maximum principle is satisfied. That is,  $\frac{\partial \mathcal{H}}{\partial \vec{u}} = 0$  if no constraint is binding and  $\delta \vec{u} = 0$  if a constraint is binding (otherwise, either the maximum principle would be violated or  $\vec{u}$  would be inadmissible). Thus,

$$\delta J = \vec{\psi}^T(t_1) \delta \vec{x}(t_1) - \vec{\psi}^T(t_0) \delta \vec{x}(t_0) \quad (6)$$

Equation (1) follows immediately from this result. In a similar fashion, it may be shown from Eq. (4) that

$$\left. \begin{aligned} \frac{\partial J}{\partial t_1} \Big|_{\delta \vec{x}_1 = 0} &= -\mathcal{H}(\vec{\psi}(t_1), \vec{x}_1, t_1) \\ \frac{\partial J}{\partial t_0} \Big|_{\delta \vec{x}_0 = 0} &= \mathcal{H}(\vec{\psi}(t_0), \vec{x}_0, t_0) \end{aligned} \right\} \quad (7)$$

Comments. - (a) Having shown that  $J = J(\vec{x}_0, \vec{x}_1)$  and computed its gradient as per Eq. (1), the variational transversality conditions now follow immediately from the well known necessary conditions for ordinary maxima and minima. At the terminal manifold  $\mathcal{M}_1$  for instance, feasible perturbations of  $\vec{x}_1$  must (to the first order) lie in the tangent plane to  $\mathcal{M}_1$  at  $\vec{x}_1$ . Clearly, for  $J(\vec{x}_0, \vec{x}_1)$  to be stationary it is necessary that

$$\vec{\psi}(t_1)^T \vec{T}_1 = 0 \quad (8a)$$

for all vectors  $\vec{T}_1$  tangent to  $\mathcal{M}_1$  at  $\vec{x}_1$ , and similarly

$$\vec{\psi}(t_0)^T \vec{T}_0 = 0 \quad (8b)$$

at  $\mathcal{M}_0, t_0$  and  $\vec{x}_0$ .

(b) The adjoint vector  $\vec{\psi}(t)$  also represents the outward normal to  $\mathcal{A}'(\vec{x}_0, t_0; t)$ , the set of attainable states defined in the  $N + 1$  - dimensional space  $[J, x_1 \cdots x_N]$  - c.f., reference 137.

This fact provides a geometrical interpretation of the preceding results and a means of understanding the role of the second-order hypotheses of the theorem. The notation and precise definition for the sets of attainability are given in Section 8.1.1.

These concepts are illustrated in Figure 3-3, where  $x_0 = J$  is plotted against a typical state variable  $x_i(t_1)$ . The set  $\mathcal{A}'(\vec{x}_0, t_0; t_1)$  and its boundary are shown in the upper part of the figure. It's projection into state space,  $\mathcal{A}(\vec{x}_0, t_0; t_1)$  is shown edge-on as the heavy line along the  $x_i$ -axis. The adjoint vector  $\vec{\psi}(t_1)$  and tangent plane  $\pi(t_1)$  are shown at the terminal point  $x_i(t_1)$ . By optimality,  $\vec{x}(t_1) \in \partial \mathcal{A}'$  and, to the first order, optimal perturbations lie in  $\pi$ . Clearly, if  $\partial \mathcal{A}'$  has the smooth, regular structure implied by the figure, the first order change  $\delta J$  due to the perturbation  $\delta x_i(t_1)$  is given by

$$\delta J = - \frac{\psi_i(t_1)}{\psi_0} \delta x_i(t_1) \left( \text{or, } \delta J = + \frac{\psi_i(t_0)}{\psi_0} \delta x_i(t_0) \right)$$

Choosing  $\psi_0 = -1$  as a scale factor and passing to the limit  $\delta x_i \rightarrow 0$ , we recover Eq. (1).

Evidently this result depends only upon  $\partial \mathcal{A}'$  having a well-behaved structure. And, although a rigorous discussion is not intended, the second-order hypotheses of the theorem actually imply proper structure of  $\partial \mathcal{A}'$ . Specifically, the continuity properties

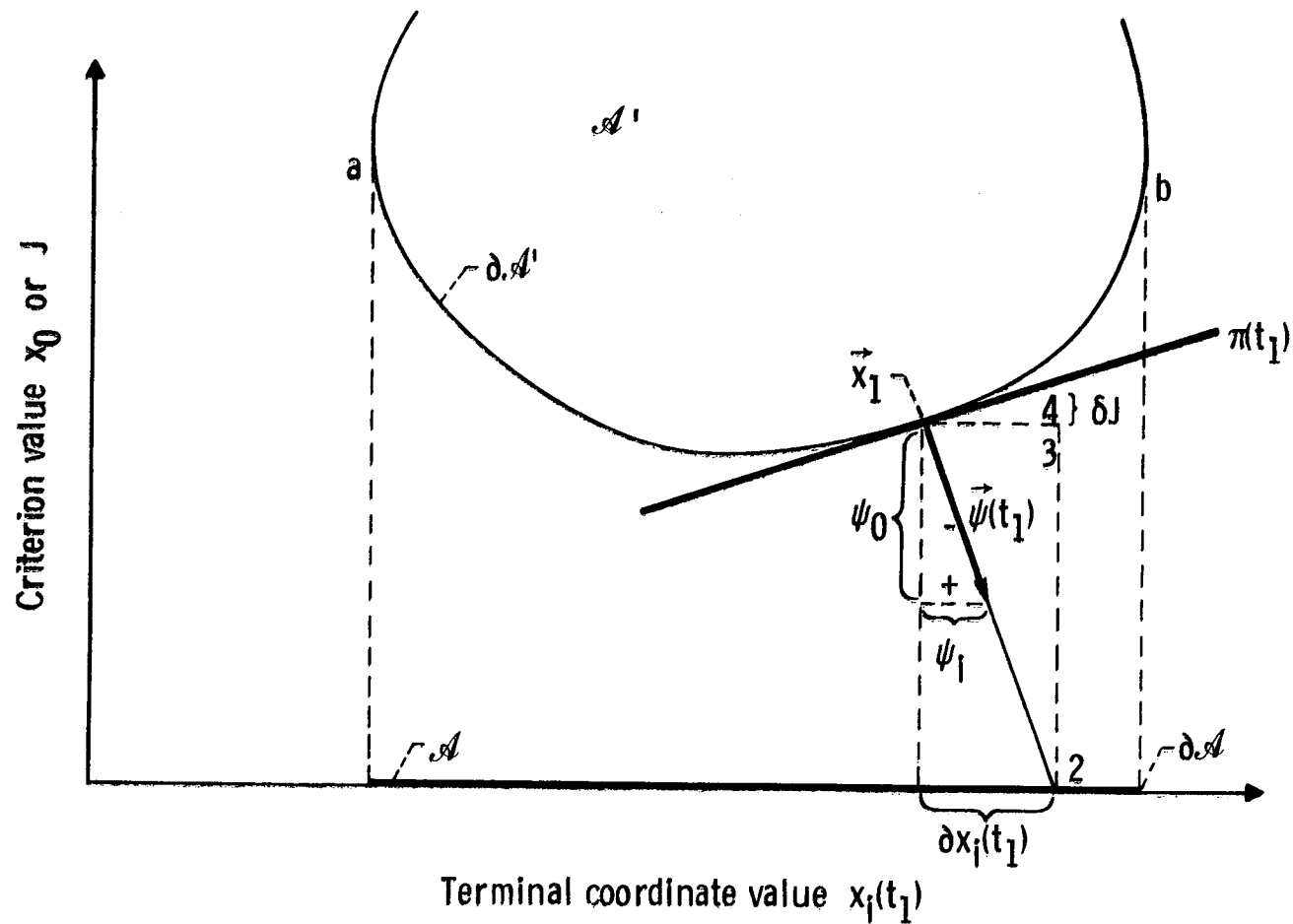


FIGURE 3-3. - GEOMETRY OF THE ATTAINABLE SETS  $\mathcal{A}'_N \subset \mathbb{R}_{N+1}$ , AND  $\mathcal{A} \subset \mathbb{R}_N$ .

together with the convexity condition mean that  $\partial \mathcal{X}'$  does not have

"corners" or other irregularities where  $\vec{\psi}(t)$  would be undefined.

The normality condition means that  $\vec{x}(t_1) \notin \partial \mathcal{X}$  (c.f., points "a"

and "b", at which  $\psi_0 = 0$ ); and hence that the scale factor choice

$\psi_0 = -1$  is legitimate. Finally, the Jacobi condition implies that

$\partial \mathcal{X}$  is not "folded" or multi-sheeted in a neighborhood of  $\vec{x}_1$ , thus establishing uniqueness in a local sense.

### 3.2.2 Necessary Conditions in $\mathcal{X}_{pp}$

For an individual arc the minimum criterion value  $J_{k,min}$  is unique if it exists at all, and it depends only on the end-point coordinates as shown above. Hence, the original integral criterion may be expressed as

$$J = \sum_{k=0}^{K-1} J_{k,min}(\vec{x}_k; \vec{x}_{k+1}) \quad (1)$$

Thus, the upper level problem reduces to a problem in ordinary calculus or mathematical programming, namely:

$$\text{minimize } J(\vec{x}_0; \vec{x}_1 \cdots; \vec{x}_K) \quad (2)$$

where

$$\vec{x}_k \in \mathcal{X}_{pp}$$

If  $\mathcal{X}_{pp}$  is the whole space  $R_{NK}$  the classical necessary conditions

$$\frac{\partial J}{\partial \vec{x}_k} = 0; \quad K \triangleq \begin{bmatrix} \frac{\partial^2 J}{\partial \vec{x}_k^2} \end{bmatrix} \quad \begin{matrix} \text{positive} \\ \text{definite} \end{matrix} \quad (3)$$

apply to the problem thus stated.

More generally, the celebrated Kuhn-Tucker conditions, c.f., reference 45, would apply if  $\bar{X}_{pp}$  were a general convex subset of  $R_{NK}$ . But for convenience, let us first consider the case where the points in  $\bar{X}_{pp}$  are unconstrained.

Unconstrained mesh points. - In view of theorem 3.2.1, the above first partial derivatives are given by

$$\frac{\partial J}{\partial \vec{x}_k} = + \vec{\psi}(t_k^+) \Big|_{\text{arc}_{k-1}} - \vec{\psi}(t_k^-) \Big|_{\text{arc}_k} \quad (4)$$

In  $R_{NK}$  we may, therefore, write the gradient vector as

$$-\nabla_X J = \begin{bmatrix} \vec{\psi}(t_0^-) - 0 \\ \vec{\psi}(t_1^-) - \vec{\psi}(t_1^+) \\ \vdots \\ 0 - \vec{\psi}(t_K^+) \end{bmatrix} \quad (5)$$

At a given point in time, the components of the gradient of  $J$  are defined in the space of mesh points; hence the name, "mesh gradient", for the present concept. Assuming that  $B_k^{-1}$  exists, the Hessian or second-partial matrix  $K$  may be determined by re-arranging Eq. 3.1

(3) in the form

$$\begin{bmatrix} \delta \vec{\psi}(t_k) \\ \delta \vec{\psi}(t_{k+1}) \end{bmatrix} = \begin{bmatrix} E & F \\ G & H \end{bmatrix} \begin{bmatrix} \delta \vec{x}(t_k) \\ \delta \vec{x}(t_{k+1}) \end{bmatrix} \quad (6)$$

where

$$\begin{aligned} E_k &= -B_k^{-1} A_k & F_k &= B_k^{-1} \\ G_k &= C_k - D_k B_k^{-1} A_k & H_k &= D_k B_k^{-1} \end{aligned} \quad (7)$$

Then by making the following substitutions: (a) 3.1 (6) into (7); (b) (7) into (6); and (c) (6) into (5); collecting terms, and formally differentiating the result, it can be shown that  $K$  has the block-tridiagonal form

$$K = \begin{bmatrix} -E_0 & -F_0 & & & & \\ \hline G_0 & H_0 - E_1 & -F_1 & & & \\ \hline & G_1 & H_1 - E_2 & -F_2 & & \\ & & \ddots & \ddots & \ddots & \\ & & & G_{K-2} & H_{K-2} - E_{K-1} & -F_{K-1} \\ & & & & G_{K-1} & H_{K-1} \end{bmatrix} \quad (8)$$

Equations (5) and (8) may be used to verify the necessary conditions, such as (3), which apply to the upper level.

P.V.C Terms. - Since  $J$  is in general a sum of integral and point-value contributions, the general gradient is obtained by adding a term such as  $\frac{\partial p_k(\vec{x}_k)}{\partial \vec{x}_k}$  to each element of the gradient shown in Eq. (5). That is, if  $p_k = p_k(\vec{x}_k)$  only, then

$$-\nabla_{\mathbf{X}} J = \begin{bmatrix} \vec{\phi}_0 \\ \vdots \\ \vec{\phi}_K \end{bmatrix} = \Psi \in R_{NK} \quad (9)$$

where

$$\left. \begin{aligned} \vec{\phi}_0 &= -\vec{\psi}_0 + \frac{\partial p_0(\vec{x}_0)}{\partial \vec{x}_0} \\ \vec{\phi}_k &= -\vec{\psi}_k + \vec{\psi}_k^+ + \frac{\partial p_k(\vec{x}_k)}{\partial \vec{x}_k} \\ \vec{\phi}_K &= \vec{\psi}_K^+ + \frac{\partial p_K(\vec{x}_K)}{\partial \vec{x}_K} \end{aligned} \right\} \quad (10)$$

and

Similarly, a term such as  $\frac{\partial^2 p_k}{\partial \vec{x}_k^2}$  must be added to each diagonal block in the second derivative matrix of (8).

In the more general case where the  $k^{\text{th}}$  PVC is influenced by mesh points other than  $\vec{x}_k$ , the additive terms indicated above are replaced by summation over the appropriate indices. That is, if  $k^*$  represents the range of indices for which  $p_{k^*}$  is influenced by  $\vec{x}_k$ , the general gradient term is

$$\vec{\phi}_k = \vec{\psi}_k^+ - \vec{\psi}_k + \sum_{k^*} \frac{\partial p_{k^*}}{\partial \vec{x}_k} \quad (11)$$

The second-derivative matrix  $K$  is similarly modified, keeping in mind that an off-diagonal block will be created each time  $k^* \neq k$ .

Constrained mesh points. - A generalization of the transversality condition may be derived by noting that any feasible perturbation  $\delta \vec{x}_k$  must be in the hyperplane  $\pi_k$ , tangent to  $\mathcal{U}_k$  at the point  $\vec{x}_k$ . I.e.

$$\vec{\phi}_k \cdot \vec{T}_k = 0 \quad (12)$$

for all linearly independent vectors  $\vec{T}_k$  in  $\pi_k$ . That is, the

gradient components associated with  $\vec{x}_k$  must be collinear with the gradient to  $\eta_k$  at  $\vec{x}_k$ . More general conditions derived from Kuhn-Tucker theory are presented in Appendix B. From henceforth, it will be assumed that all these conditions have been enforced, so that the gradient is understood to be taken with respect to feasible directions only.

### 3.2.3 Direction-of-Descent Algorithms

Having now developed the structure of the criterion value function and the necessary conditions that apply to its minimization, it is finally appropriate to consider numerical algorithms for implementing Level II. There are many which could be considered, for instance references 44, 45, 48, 49, 57-60, 67, 68, 83, 87, 122, 138, 176, and 185. Three algorithms of present interest are presented in Appendix B and their characteristic features are developed. These are:

- (a) An apparently - novel, "maximally smooth" linear step algorithm;
- (b) A second-order step algorithm; and
- (c) A modification of the Fletcher-Powell algorithm in which the unidimensional-search step length is computed in closed form from second-order data (as opposed to numerical linear searching)

These algorithms differ significantly in their convergence rates and other details, but all function by defining and searching along

successive directions of descent in  $\mathbf{X}_{pp}$ . Hence, it is of interest to consider the convergence properties of direction-of-descent algorithms as a class. In the present terms it is possible to define candidate directions of descent as

$$\hat{s} = \{\text{unit vectors in } \mathbf{X} \mid \hat{s} \cdot \Psi < 0\} \quad (1)$$

That all such directions are directions of descent may be shown as follows. Suppose that the  $NK \times NK$  second partial matrix

$$\frac{\partial^2 J(\mathbf{X}_*)}{\partial \mathbf{X}_i \partial \mathbf{X}_j} = \nabla_{\mathbf{X}}^2 \Psi(\mathbf{X}_*) \quad (2)$$

is well defined and has a uniform bound in some neighborhood of the vector

$$\begin{bmatrix} \vec{x}_1^* & \dots & \vec{x}_K^* \end{bmatrix}^T = \mathbf{X}_*$$

(By uniform bound, it is meant that there is a number  $m > 0$ , such that

$$|\vec{z}^T [\nabla_{\mathbf{X}} \Psi(\mathbf{X}_*)] \vec{z}| \leq m |\vec{z}|^2 \quad (3)$$

for all  $\vec{z} \in R_{KN}$ .)

Then, use Taylor's theorem with remainder to express the value of  $J$  resulting from a step of length  $\alpha > 0$  in the  $\hat{s}$ -direction, from  $\mathbf{X}_*$ :

$$J(\mathbf{X}_* + \alpha \hat{s}) = J(\mathbf{X}_*) + \alpha \hat{s}^T \nabla_{\mathbf{X}} J + \frac{1}{2} \alpha^2 \hat{s}^T [\nabla_{\mathbf{X}}^2 \Psi(\xi)] \hat{s} \quad (4)$$

where

$$\xi = \theta \mathbf{X}_* + (1 - \theta)[\alpha \hat{s}]$$

and

$$0 \leq \theta \leq 1$$

Now, recalling that  $\hat{s}$  is a unit vector, defining  $\beta$  to be the negative of the cosine of the angle from  $\hat{s}$  to  $\Psi$  and using the uniform bound relation we get

$$J(\mathbf{X}_* + \alpha \hat{s}) \leq J(\mathbf{X}_*) + \alpha \beta |\nabla_{\mathbf{X}} J| + \frac{\alpha^2 m}{2} \quad (5)$$

Since  $\beta < 0$  by hypothesis, it is always possible to find a sufficiently small  $\alpha > 0$  such that the right hand side of (5) is less than  $J(\mathbf{X}_*)$  (e.g., any value of  $\alpha$  in the range  $0 < \alpha < \frac{2}{m} |\beta \nabla_{\mathbf{X}} J|$ ). Hence, the vectors  $\hat{s}$  defined above are directions of descent.

A direction of finite descent is similarly defined as

$$\hat{s} = \{\text{unit vector in } \mathbf{X} \mid \hat{s} \cdot \Psi = < -\epsilon < 0\} \quad (6)$$

At this point it is intuitively clear that descent algorithms based on always searching along a direction of finite descent, will eventually converge if  $J$  is bounded below. This is of fundamental importance and is therefore stated as a formal:

Theorem. - Given the hypotheses of section 8.1.1, a uniform bound for Eq. (2) above, and that  $J$  is bounded below. Assume that an algorithm using only directions of finite descent is applied to the function  $J(\mathbf{X})$ .

Then. - The descent process will eventually converge, in the sense that  $J$  will ultimately approach its "nearest" local minimum.

Proof. - From the triangle inequality and the above mentioned bound, it is seen that

$$J(\mathbf{x}_* + \alpha \hat{s}) \leq J(\mathbf{x}_*) + \alpha \left[ \frac{\alpha m}{2} - |\beta \nabla J| \right] \quad (7)$$

where  $\alpha$ ,  $\beta$ , and  $m$  are as used above.

Arbitrarily set

$$\alpha = \frac{1}{m} |\beta \nabla J| \quad (8)$$

then

$$J(\mathbf{x}_*) - J(\mathbf{x}_* + \alpha \hat{s}) \geq \frac{|\beta \nabla J|^2}{2m} \quad (9)$$

Recalling that

$$\beta \geq \epsilon \quad (10)$$

which follows from the definition of a finite descent direction, consider the sequence  $\{\mathbf{x}_i\}$  and assume it has a limit point  $\mathbf{x}$ . Form the sums

$$\sum_{k=0}^{p-1} [J(\mathbf{x}_k) - J(\mathbf{x}_{k+1})] = J(\mathbf{x}_0) - J(\mathbf{x}_p) > \frac{1}{2m} \sum_{k=0}^{p-1} |\beta_k \nabla J_k|^2 \quad (11)$$

(Eq. (11) follows from the fact that  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \hat{s}_k$ ). Thus,

$$J(\mathbf{x}_p) \leq J(\mathbf{x}_0) - \frac{1}{2m} \sum_{k=0}^{p-1} |\beta_k \nabla J_k|^2 \quad (12)$$

Now, taking note of the fact that  $J$  is bounded below (because  $f_0$  is bounded below and  $\mathcal{X}$  is finite) and passing to the limit

$$\lim_{p \rightarrow \infty} J(\mathbf{x}_p) \leq J(\mathbf{x}_0) - \frac{1}{2m} \sum_{k=0}^{\infty} |\beta_k \nabla J_k|^2 \quad (13)$$

then shows that the series  $\sum_{k=0}^{\infty} |\beta_k \nabla J_k|^2$  converges, which means that

$$\lim_{k \rightarrow \infty} |\beta_k \nabla J_k|^2 = 0 \quad (14)$$

This implies that

$$\lim_{k \rightarrow \infty} |\epsilon \nabla J_k|^2 = 0 \quad (15)$$

because of (10), and hence

$$\lim_{k \rightarrow \infty} |\nabla J_k|^2 = 0 \quad \text{since } \epsilon > 0 \quad (16)$$

by definition. Thus, any algorithm which produces directions of descent will eventually converge.

In passing, it should be noted that the finite descent condition (6) is viewed here as an empirical, verifiable restriction on admissible search directions, not necessarily as an analytical property of particular algorithms or objective functions. The class of "gradient-restoration" techniques (c.f., ref. 122) functions by, in effect, keeping track of  $|\beta|$  and restoring the gradient direction ( $|\beta| = 1$ ) as the current search direction whenever a predetermined critical value of  $\epsilon$  is reached.

To summarize, the essential Level II operations are to compute the criterion value  $J(\vec{x}_0, \vec{x}_1 \cdots \vec{x}_K)$  the gradient vector (3.2.2(5)), the Hessian matrix  $K$  (3.2.2(8)) and then apply a direction-of-descent algorithm to find the optimum mesh point locations in  $\mathbf{x}_{pp}$ . The overall MG method then entails the following major steps.

(a) An initial set of mesh points is selected by judgment, observing the criteria of 2.4(c) and 3.1.3.

(b) The resulting sequence of optimal TPBVP's is solved using the TMA (summarized at the end of Section 3.1.2), or any other

appropriate method. This comprises one cycle of Level I and produces the data needed in Level II for step  $i$ .

(c) Level II then computes the quantities  $J_i$ ,  $\nabla_{\mathbf{x}} J_i$  and  $K_i$ , and applies one of several appropriate search algorithms to define a direction of descent  $\hat{s}_i$  in  $\mathbf{X}_{pp}$  and a step length  $\alpha_i$  such that

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \hat{s}_i$$

$J_{i+1}$  is minimized with respect to  $\alpha_i$ , and

$$J_{i+1} < J_i \quad \text{if } \alpha_i > 0$$

(d) The necessary conditions applicable to Level II, c.f., Eq. 3.2.2(3) are checked at each step. Steps (b) through (d) are repeated until 3.2.2(3) is satisfied within acceptable tolerances.

#### 4. SAMPLE APPLICATIONS

To illustrate the previous developments, the Mesh Gradient approach has been applied to four sample problems.

Example 1, a minimum effort control problem with field-free dynamics, is intended merely to illustrate the Mesh Gradient method, using several different descent techniques.

Example 2, involves a conjugate-point problem previously analyzed in reference 24, and illustrates the present method's strong tendency to reject conjugate solutions.

Example 3, the optimal control of an unstable Van Der Pol oscillator, is pursued in considerable detail to demonstrate the efficiency of the method for highly unstable problems with numerous intermediate constraints. Comparisons with alternative methods of solution are included.

Example 4, illustrates the method's ability to efficiently solve optimal, multi-impulse and low-thrust space trajectory problems and develops the existence of a hitherto-unsuspected class of solutions.

##### 4.1 Example 1 - Minimum-Effort Control

As a simple illustrative example, consider the following problem:

$$\text{Minimize } J = \frac{1}{2} \int_0^{10} u^2(t) dt \quad (1)$$

with

$$\dot{x} = u; \quad x(0) = x(10) = 0$$

Obviously, we have that  $U_{\text{opt}} = \psi$ , and  $\dot{\psi} = 0$ . Hence, optimal trajectories consist of straight lines, with slopes given by the local value of  $\psi$ , between successive terminals. The components of the vector  $\Delta\psi$  referred to previously, are given in this problem simply by the change in slope at each interior mesh point.

For the sake of being definite, let there be 11 equally spaced mesh points, initially located as follows:

$$\begin{aligned} x_0^0 &= x_{10}^0 = 0 \\ x_i^0 &= 1, \quad 1 \leq i \leq 9 \end{aligned} \quad (2)$$

The corresponding values of  $\Delta\psi_i$  are all zero except for

$$\Delta\psi_1^0 = \Delta\psi_9^0 = -1.$$

#### 4.1.1 Descent Via Programming

Proceeding by steepest descent we obtain the sequence illustrated below. (Note that because of symmetry about  $t = 5$ , we have  $x_4 = x_6$ , etc..) In Figure 4.1, the initial guess  $\vec{x}^0$  and the corresponding  $\Delta\vec{\psi}^0$  are shown in heavy lines, the succeeding state-iterates are indicated by light lines and identified by step number. ( $\Delta\vec{\psi}_1$ , but no further  $\Delta\vec{\psi}_k$ 's is shown.) It is apparent from the figure that convergence is very slow. That is, the error measure  $\max_{t,k} |\vec{x}_k(t) - \vec{x}_{\text{opt}}(t)|$  is not reduced appreciably until 9 steps have been taken (or in general,  $K$  steps if  $K$  rather than 9 interior mesh points had been used).

This slow convergence is a well-known characteristic of the gradient method. However, if the conjugate-gradient algorithm is used

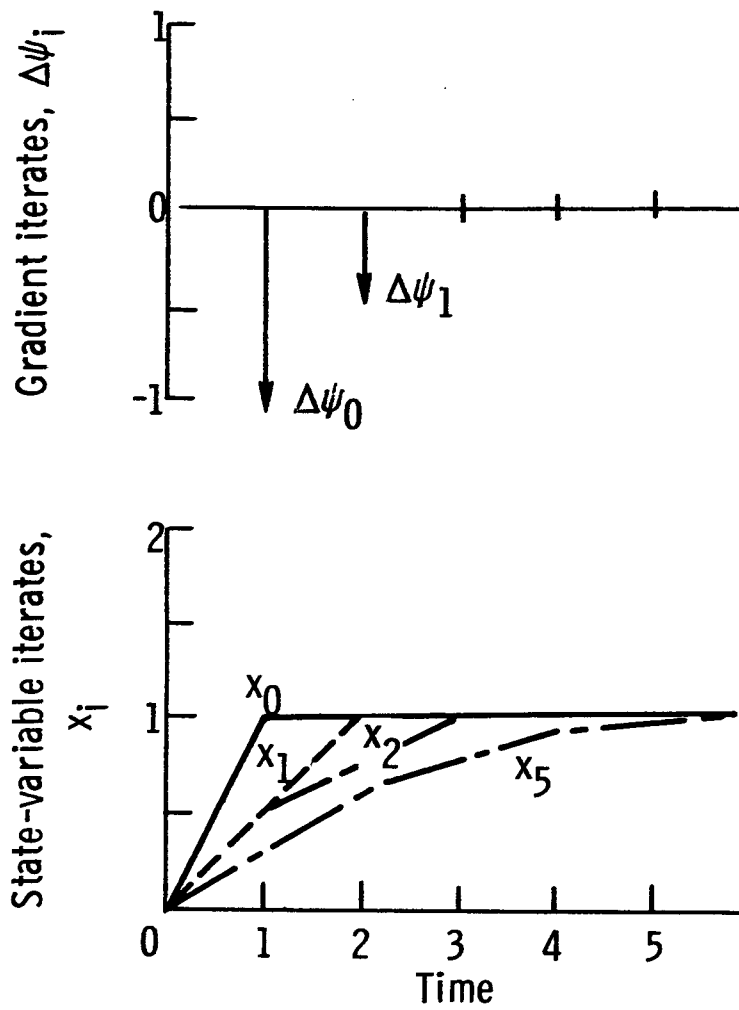


FIGURE 4-1. - INITIAL-GUESS DATA  
FOR PROBLEM 4.1.

instead, exact convergence is attained after 5 steps (or about  $K/2$  steps generally), as Table 4-1 shows.

Table 4-1  
Conjugate Gradient Step Data for Problem 4.1

Step	0	1	2	3	4	5
J	1	1/2	1/3	1/4	1/5	0
$x_1$	1	1/2	1/3	1/4	1/5	0
$x_2$	1	1	2/3	1/2	2/5	0
$x_3$	1	1	1	3/4	3/5	0
$x_4$	1	1	1	1	4/5	0
$x_5$	1	1	1	1	1	0
$s_1$	-1	-1/4	-1/9	-1/16	-2/25	0
$s_2$	0	-2/4	-2/9	-2/16	-4/25	0
$s_3$	0	0	-3/9	-3/16	-6/25	0
$s_4$	0	0	0	-4/16	-8/25	0
$s_5$	0	0	0	0	-10/25	0
Symmetrical about $x_5, s_5$ .						
.	.	.	.	.	.	.

The preceding transparently simple example does more than merely demonstrate one method of applying state-gradient theory to optimization problems; it also illustrates the primary disadvantage of that method. I.E. because of the fixed boundary points, the initial step-lengths are relatively small and convergence is correspondingly delayed. It is clear from Figure 4-1 especially that there is no significant reduction of  $\max_{t,j} |x_j(t) - x_{opt}(t)|$  until the boundary-point influences have propagated clear across the trajectory.

#### 4.1.2 Descent by First or Second Order Step

Anticipation of the above difficulties led, in Section 9.3.1 of Appendix B, to an attempt at constructing a direction of descent  $s$ , which vanishes in a smooth, continuous fashion at the fixed terminals.

For the same problem and initial iterate considered above, the  $g$ ,  $h$ ,  $s$ , and  $\Delta\psi$  functions of Section 9.2.1 may be computed as follows:

$$j = 1$$

$$h = \dot{x}$$

$$\frac{\partial h}{\partial x} = 0 \quad \frac{\partial h}{\partial t} = 0$$

$$\frac{\partial h}{\partial \dot{x}} = 1$$

$x^*(t)$ : as above with

$$\ddot{x}^* = -\delta(t-1) - \delta(t-9)$$

$$g = 0$$

$$\therefore \Delta\psi = +\delta(t-1) + \delta(t-9)$$

Hence, search directions are defined by

$$\ddot{s} = \lambda(\delta(t-1) + \delta(t-9))$$

$$\dot{s} = +\lambda(u(t-1) + u(t-9)) + a$$

$$s = +\lambda(r(t-1) + r(t-9)) + at + b$$

Here,  $\delta(\ )$ ,  $u(\ )$  and  $r(\ )$  denote the unit impulse, step and ramp function of the indicated argument;  $a$  and  $b$  are constants of integration. Clearly  $b = 0$ ,  $a = -\lambda$ , hence

$$s = \lambda(-r(t) + r(t-1) + r(t-9))$$

With  $\lambda = 1$ ,  $s$  is precisely the negative of  $x_0$ , and the optimum,  $x(t) \equiv 0$ , is attained in one step. It is apparent that one-step convergence will always be attained by this method in the special case where the state equations are field-free (the  $x_1$  not appearing at all) and linear in the control. The second-order step method will obviously yield equivalent results for this particular example. It will clearly provide one-step convergence for a general linear state, quadratic criterion system, since the variational equations of Section 9.2.1 are then collectively equivalent to the "exact" transfer matrix of the linear system.

#### 4.2 Example 2 - A Problem with Conjugate Points

As initially stated, the preceding example serves merely to illustrate the application of the Mesh Gradient technique with several descent techniques. Before proceeding to more complex examples it is perhaps of interest to provide a simple demonstration of one of the method's more important qualitative features - namely, its ability, for some class of problems, to reject conjugate-point solutions.

A conjugate-point problem which may be solved analytically appears in Bryson & Ho (ref. 24); it is: Minimize the time required to reach  $(x_f, 0)$  from the origin for the system

$$\dot{x} = V \cos \theta$$

$$\dot{y} = V \sin \theta$$

where the scalar velocity is

$$V = V_0 \sqrt{1 + y^2/h^2}$$

and  $\theta$  is the scalar control direction.

Upon introducing the variational adjoint function and Hamiltonian, replacing  $t$  by  $x$  as the independent variable and using the maximum principle, we obtain the canonical system

$$\frac{dy}{dx} = y' = \tan \theta \quad J' = \frac{\sec \theta}{v} \quad (1)$$

$$\theta' = \frac{-y}{(h^2 + y^2)}$$

with boundary conditions

$$y(0) = 0 \quad y(x_f) = 0 \quad (2)$$

In deriving equations (1), use was made of the optimal control law

$$\psi = -\tan \theta [\cos \theta / v] \quad (3)$$

where the bracketed term is constant by virtue of Snell's law. Bryson & Ho showed that this problem can be solved explicitly in terms of standard elliptic integrals, and their solution (which did not include the heavy "envelope" curves) is presented as Figure 4-2(a). This figure shows some of the minimum time trajectories and contours of constant transfer time. Clearly there is a conjugate point at  $\frac{x}{h} = \pi$ ,  $y = 0$ ; the trajectory defined by  $\theta = 0$ , is optimum for  $0 < \frac{x_f}{h} \leq \pi$  but not for  $\frac{x_f}{h} > \pi$ .

Note, that the contours of constant  $\frac{v_0 t}{h}$  on the sketch develop an infinite curvature at the conjugate point (i.e.,  $\frac{\partial^2 J}{\partial y^2} \rightarrow \infty$ ). Furthermore, on  $y = 0$  beyond the conjugate point  $\left(\frac{x}{h} > \pi\right)$ , the contours of constant  $\frac{v_0 t}{h}$  have a discontinuity in slope upon crossing the X-axis.

Reproduced from  
best available copy.

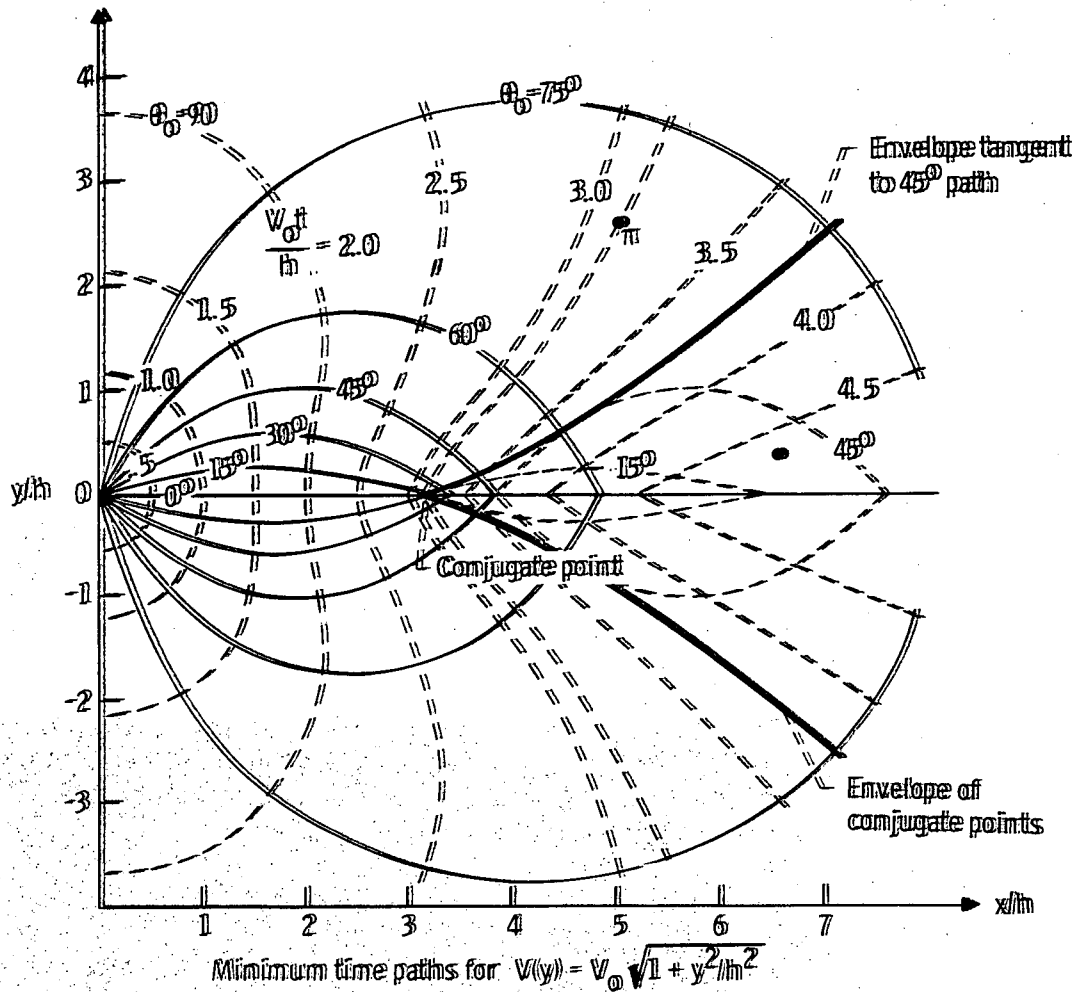


FIGURE 4-2(A). - SOLUTION CURVES FOR PROBLEM 4.2.

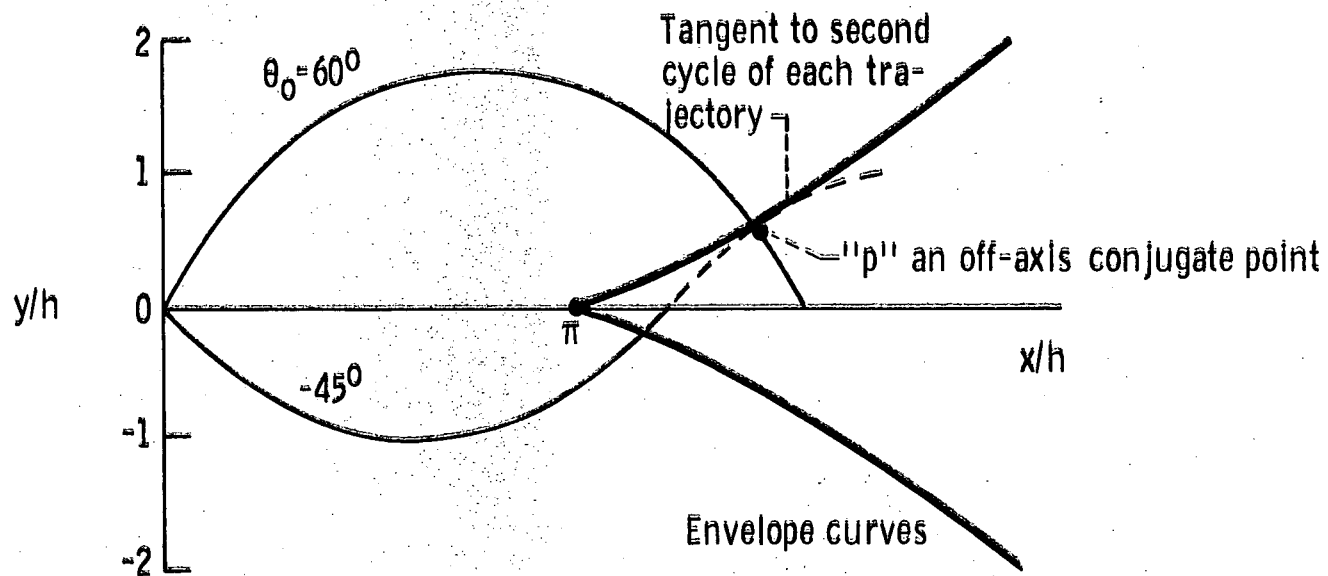


FIGURE 4-2(B). - ILLUSTRATION OF CONJUGATE POINT.

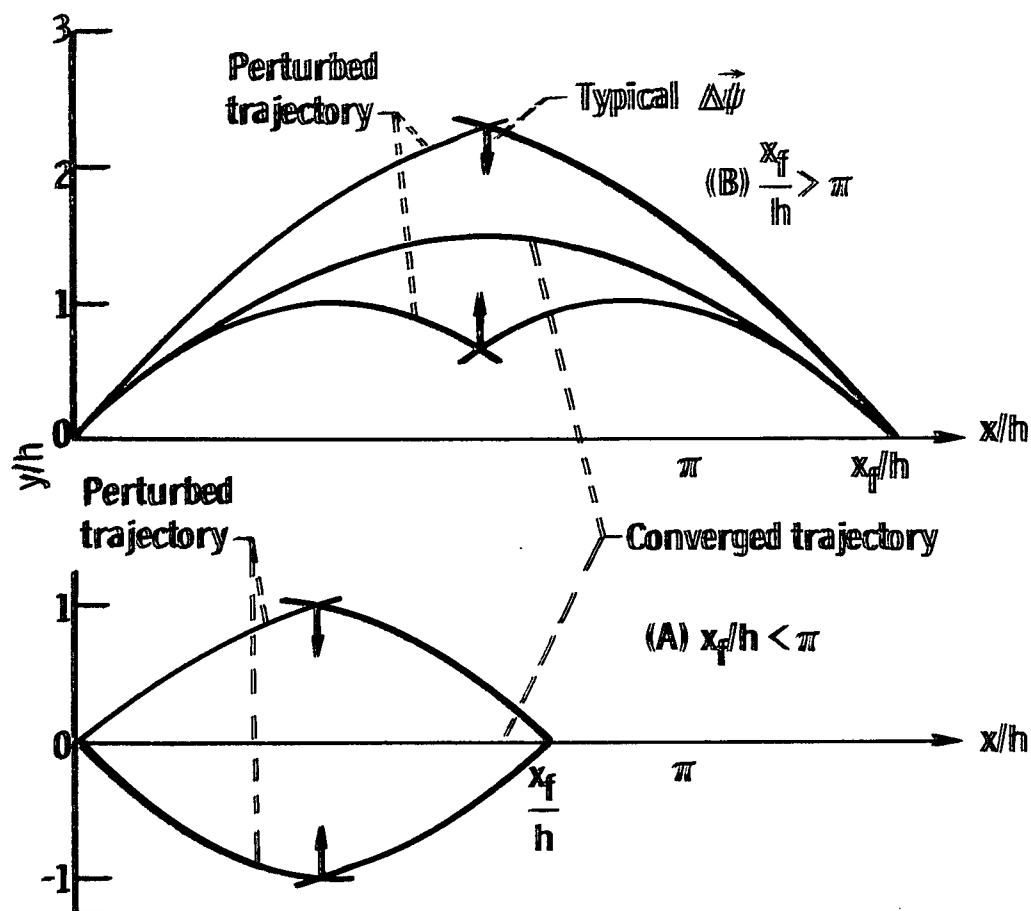
Although this complete solution is already available, it seems that some further discussion might be instructive. Note, first, that the conjugate point appearing at  $x_f/h = \pi$  is actually the "leading edge" of a cusp-shaped region filled with conjugate points. The envelopes delimiting this region are shown by the heavy black curves in the figure. Points within this region may be reached in at least two ways. This may be understood when it is realized that the solution curves for each value of  $\theta_0$  are periodic; so that, a given point "P" may be reached either directly or by a trajectory containing one or more intermediate zero-crossings. See Figure 4-2(b).

Of perhaps greater interest is the fact that the Mesh Gradient approach appears to be unaffected by the presence of these conjugate points if the mesh spacing is taken to be  $\frac{\Delta x}{h} < \pi$ . This may be understood with the aid of Figure 4.3, when it is recalled from Eq. (3) that

$$\psi \propto -\tan \theta \quad (4)$$

Let us assumed, for simplicity, that a single, symmetrically spaced mesh point is imposed at  $x_1 = x_f/2$ . The perturbed trajectory is constructed by running arc 1 from  $(0,0)$  to  $(y_1, x_1)$  and arc 2 from  $(0, x_1)$  back to  $(y_1, x_1)$ . The gradient of  $J$  at  $(y_1, x_1)$  is clearly proportional to  $2 \tan \theta_1$ . Thus,  $J$  will decrease when  $y$  moves into the acute angle formed by the 2 arcs. If  $x_f < \pi$ , the acute angle is always on the side nearer to the X-axis (Fig. 4-3(a)) so that proper numerical searching will then select  $y \equiv 0, \theta \equiv 0$  as the optimal trajectory.

Reproduced from  
best available copy.



**FIGURE 4-3. - DIRECTIONS OF DESCENT FOR PROBLEM 4.2.**

On the other hand, if  $\frac{x_f}{h} > \pi$ , (Fig. 4-3(b)) the acute angle is initially away from the X-axis, if  $y_1/h$  is small. Hence, searching will proceed in the direction of increasing  $y_1/h$  until an optimum height is found.

Although the MG technique's ability to reject conjugate solutions for some problems is recognized as a valuable feature, it clearly does not hold true for all problems and all perturbations. On the surface of a cylinder, for instance, there are two geodesic curves joining each distinct pair of points, and in contrast to the example above, these cannot be embedded in a single, continuous family of varied trajectories. On the other hand, the Jacobi condition may be verified for individual arcs as shown in Section 8.2.3 and 8.2.4. Moreover, the condition can be verified for a total problem if the overall transition matrix is known. (It may be found by merely forming the product of subarc transition matrices.) The resulting A, B, C, and D blocks may then be used to verify conditions 8.2(2) and 8.2(3).

#### 4.3 Example 3 - Control of an Unstable Van Der Pol Oscillator

The basic computational steps of the Mesh-Gradient method, outlined in Chapter 3, were applied to the following sample problem.

$$\text{Minimize } J = \int_{t=0}^{10} (x_1^2 + x_2^2 + u^2) dt \quad (1)$$

where

$$\dot{x}_1 = (1 - x_2^2)x_1 - x_2 + u; \quad \dot{x}_2 = x_1;$$

$$x_1(0) = 0; \quad x_1(10) \text{ is open};$$

$$x_2(0) = 3; \quad x_2(10) \text{ is open}$$

and  $u$  is the scalar, unconstrained control variable. These equations represent a nonlinear oscillator which displays limit cycle behavior in the absence of control. The reasons for selecting this example were that:

- (a) Its low dimensionality facilitated checking, debugging and modifications;
- (b) By virtue of its nonlinearity and instability it provides a realistic exercise for comparing optimization methods; and
- (c) It allows direct comparison to be made with other techniques because it was also considered by Lasdon et al (ref. 100) and Mitter (ref. 123) in their studies of control-iteration methods.

#### 4.3.1 Computing Codes

By introducing adjoint variables  $\psi_1$  and  $\psi_2$  and defining the Hamiltonian function as

$$\mathcal{H} = \psi_1 \left[ (1 - x_2^2 x_1) - x_2 + u \right] + \psi_2 x_1 - (x_1^2 + x_2^2 + u^2) \quad (3)$$

the steps called for in Chapter 3 may be carried out straight forwardly. That is, the adjoint equations are found by differentiating Eq. (3), and the maximum condition yields

$$u_{\text{opt}} = \frac{\psi_1}{2} \quad (4)$$

which is then used to eliminate the control from both the state and adjoint equations. The resulting forms are finally differentiated with respect to  $x_1$  and  $\psi_1$  to obtain the differential equations defining the transfer-matrix elements.

#### 4.3.1.1 Descent by Modified Fletcher-Powell Program

The first option involves using a modified Fletcher-Powell technique to guide the upper-level search and a linearized initial-value method for the lower-level boundary problems. The organization of the code is indicated by Figure 4-4(a) and outlined in Section 9.5. A more-or-less self-explanatory program listing is available from the author. The Fletcher-Powell routine, "FLPL", was adapted from one furnished by Prof. J. D. Schoeffler (CWRU). The modification referred to above consisted mainly of using Eq. (9.4(4)) to predict the descent step length (by means of subroutine "STEPL"). (The undimensional numerical search capability of the original routine was retained as a backup feature, however.)

#### 4.3.1.2 Descent by Second-Order Steps

Also studied, was a version of the MG method in which a second order step was computed as per Section 9.3. The organization of the code is illustrated in Figure 4-5, and the program listing is also available from the author. Recall that the matrix inversions required for lower-level purposes serve double duty here; only one more  $N \times N$  matrix inversion is required to accomplish the recursive upper-level solution.

#### 4.3.1.3 Generating Initial Guesses for Lower-Level Iteration

Both codes contain a subroutine, "GUESS", whose purpose is to predict the change in the initial  $\psi$  values associated with each descent step. This keeps the lower level functioning very



**FIGURE 4-4. - PROGRAM ORGANIZATION, FLETCHER-POWELL METHOD.**

Reproduced from  
best available copy.

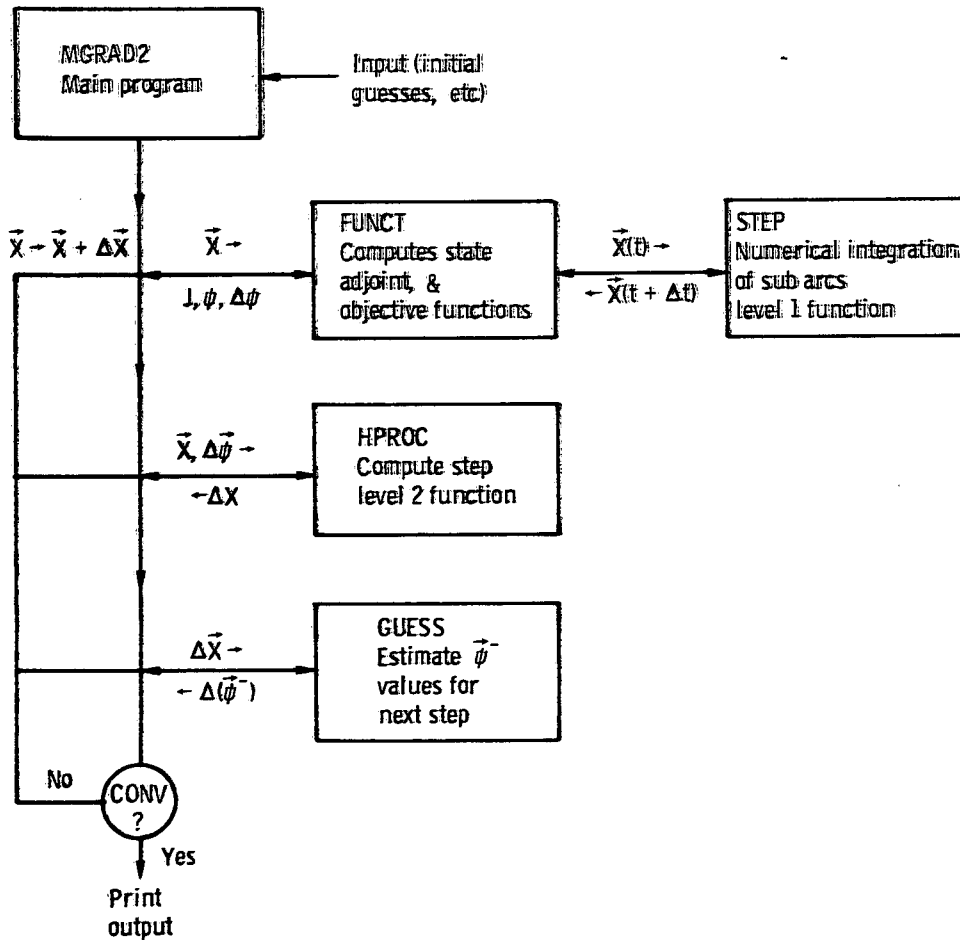


FIGURE 4-5. - PROGRAM ORGANIZATION, SECOND ORDER METHOD.

efficiently once an initial feasible case has been found. The following is a systematic way of generating an initial feasible case:

Differentiate the state equations\* once with respect to time and eliminate appearances of adjoint derivatives (such as  $\dot{\psi}_1$ ) by substituting the adjoint differential equations.\* Solve for the  $\dot{\psi}_1$  in terms of the  $x_j$ ,  $\dot{x}_j$ , and  $\ddot{x}_j$ . These values ( $\psi_1^0$ ) are the required initial guesses on the adjoint variables. They are such that the first-trial variational subarcs are tangent to the initial-estimate state trajectory\*\* at the selected partition points. For the present study, the initial feasible trajectory together with its slopes and curvatures was determined from the equations

$$\begin{aligned} x_2^0 &= 3 \exp\left(\frac{-t^2}{10}\right) \\ x_1^0 &= \frac{-6t}{10} \exp\left(\frac{-t^2}{10}\right) \end{aligned}$$

Essentially, this equation is arbitrary. It was chosen on the basis that it satisfied the boundary condition of (2) and seemed to yield a plausible type of asymptotic behavior.

#### 4.3.2 Numerical Results

The problem discussed above was programmed for numerical solutions as shown in Figures 4-4 and 4-5, and was then investigated with the aid of a digital computer. A first and general observation resulting from these computations is that the canonical equations tend to be

---

\* In which the maximum condition has been used to eliminate  $u$ .

\*\* Here assumed to be represented by the first three terms of its Taylor expansion near each mesh point, i.e.,  $x_j^0$ ,  $\dot{x}_j^0$  and  $\ddot{x}_j^0$ .

very unstable (as previously anticipated). This was due in part to the fact that the fundamental matrix, if evaluated along the entire trajectory, contained some extremely large elements (in the "B" block particularly). Additionally, nonlinear effects cause these same elements to grow with extraordinary speed as the result of perturbations. The two effects combined to make this problem completely intractable for initial-value iteration methods when it is treated as a single arc. Even when initial values resulting from previous, converged multiple arc solutions were used, the perturbations propagating from the mesh-points (due to numerical boundary-value tolerances on the order of  $10^{-6}$ ) invariably cumulated in a machine "overflow" and rendered further computation impossible. This general type of a nonlinear instability, although perhaps somewhat exaggerated in the present example, is often encountered in aerospace trajectory optimization problems. Initial trials quickly established the fact that either of the present Mesh Gradient algorithms will reduce the stability problem to insignificance if at least 10 subarcs are used. Routine solutions, even from random initial-guess trajectories, can then be made. Six subarcs are sufficient if even the slightest effort (see Eq. (6)) is made to obtain a plausible-looking initial trajectory. Using 4 subarcs or less led to persistent difficulty even when a very good initial guess was used. Five subarcs seem to represent a marginal case with results (i.e., convergence or overflow) depending upon the fine structure of the initial-guess trajectory. These considerations led to the selection of 10 subarcs as standard

for the numerical calculations reported below. It was also found that at least 30 integration steps were necessary (e.g., 3 per subarc) to produce 4 digit accuracy in the results. Except where otherwise noted, 5 steps per subarc were used herein.

#### 4.3.2.1 Comparison With Previous Methods

The critical question is whether the present approach represents any improvement over the most appropriate existing method. Since initial value methods proved quite impractical for the chosen problem it appears that some type of control iteration method should be the basis of comparison. These avoid the computational stability problem by dividing the integration into forward and return passes. The forward pass involves state variables only, using a predetermined control function; this is a stable process. The return pass consists of integrating the adjoint equation; this is stable also, because the integration is backward in time. The most recent and very probably the best methods in the control iteration category are the conjugate gradient and second variational techniques discussed in the paper by Lasdon et al (ref. 102). Some computational results from that paper are duplicated in Figure 4-6 and compared with the present Mesh Gradient results.

#### Rates of Convergence

In Figure 4-6(a), the value of  $J$  is plotted against iteration number for both the state-gradient results and the previous control iteration results. It is clear that both the Fletcher-Power and 2nd order versions of the MG approach offer more rapid convergence than

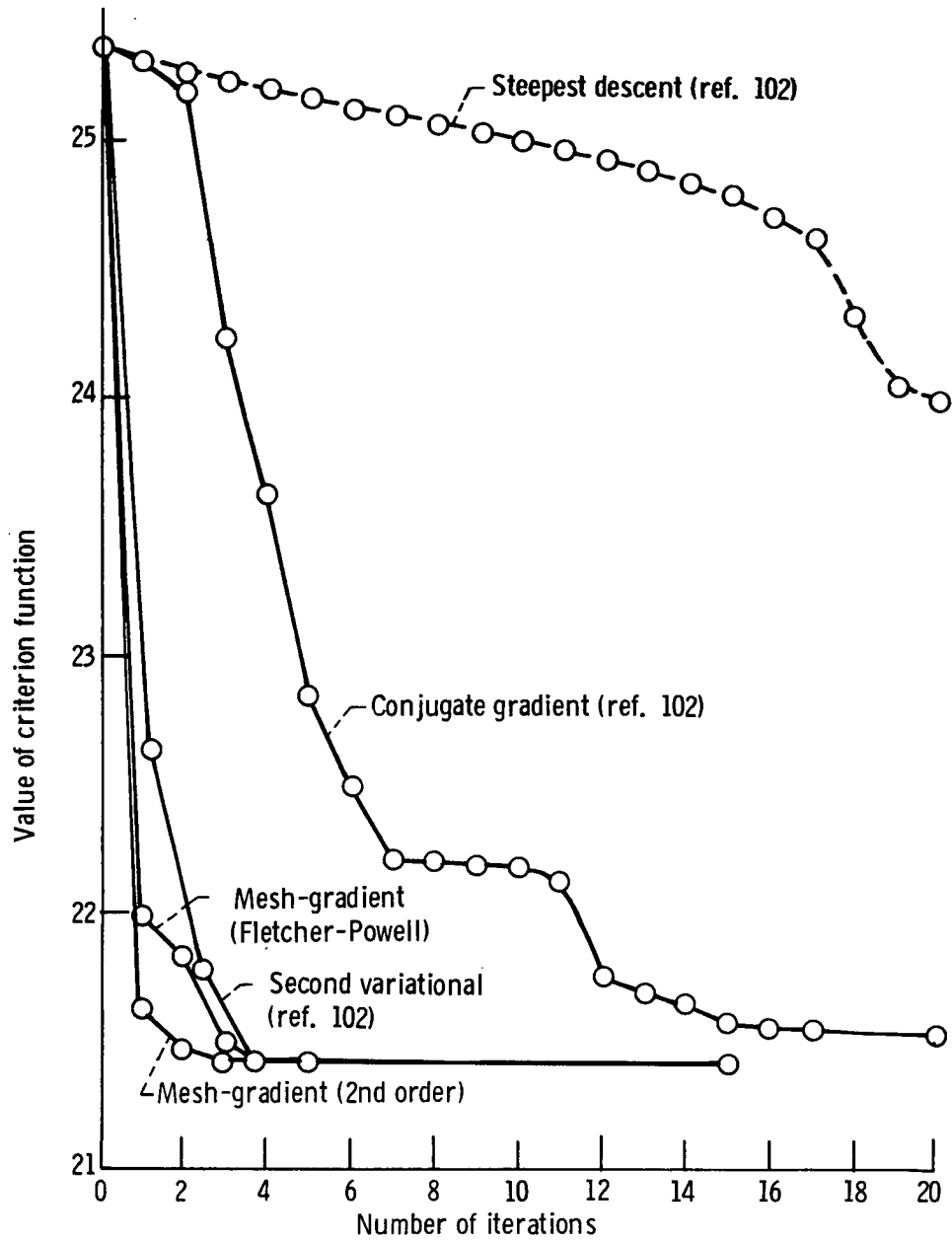


FIGURE 4-6(A). - COMPARISON OF CRITERION VALUES.

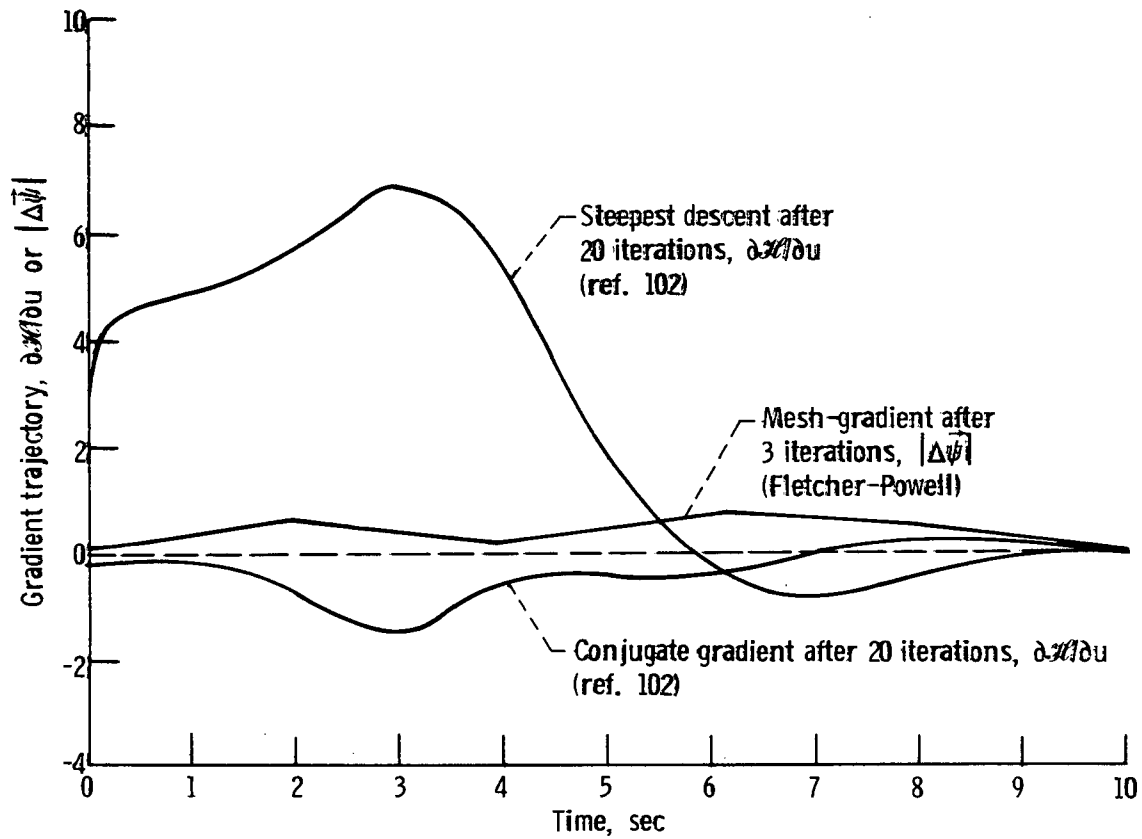


FIGURE 4-6(B). - COMPARISON OF GRADIENT TRAJECTORIES.

the control iteration methods, especially for the first step. The difference is very pronounced in comparison with the gradient and conjugate gradient methods; it is considerably smaller but probably still significant in the case of the second variational control iteration method. That is, if a tolerance of 0.5% on the value of  $J$  is accepted as defining "convergence" for practical engineering purposes, the comparison displayed in the following table is obtained:

Table 4-2  
Number of Steps Required for 0.5% Convergence

Method	Number of Steps
MG (2nd order)	2
MG (Fletcher-Powell)	3
Second Variational (Mitter)	4
Conjugate-Gradient (Lasdon)	>20 (say 30)
Steep Descent	20

The comparison is further elaborated in Figure 4-6(b), where the gradient trajectories for the conjugate gradient and MG/Fletcher-Powell methods are presented. Clearly the MG approach has produced a better state of convergence after 3 iterations than the conjugate gradient approach did after 20.

#### Computing Time

The comparison between methods is less clear-cut when performed on the basis of computer time required to reach a given degree of

convergence. Such comparisons tend to be very misleading unless great care is used to insure that identical numerical-integration methods, tolerances, and computing machines are involved in all cases. This of course, could not be done for the Figure 4-6 results. Rough estimates, however, can be made on the basis that, for a given computer, integration algorithm and number of integration steps, the time should vary as

$$N_{eq} \times N_{ds} \times N_{bv}$$

where

$$N_{eq} = \text{no. eqs. to be integrated}$$

$$N_{ds} = \text{no. of descent steps}$$

and

$$N_{bv} = \text{no. of trials to satisfy boundary values (the average value is used)}$$

For the conjugate gradient method,  $N_{eq} = 4$  (2 state and 2 adjoint equations only). For the second variational and MG methods, there are also 16 component equations of the fundamental matrix, so that  $N_{eq} = 20$ .

The values of  $N_{ds}$  were previously shown in Table 4-2. The value of  $N_{bv}$  is unity for all the control iteration methods because the terminal conditions are free for this problem. For the MG methods, it was observed that  $N_{bv}$  averaged 2 to 3 trials during the first descent step, but was limited to unity subsequently. (This means the 2nd and later steps were computed on the basis of predicted values of  $\vec{\psi}, \Delta\vec{\psi}$ , etc.). Then the applicable value of  $N_{bv}$  decreases

as  $N_{ds}$  increases; a value of 1.75 (corresponding to  $N_{ds} = 2$ ) was assumed here. The results of these estimates are presented in Table 4-3 below:

Table 4-3  
Estimated Run Times for Different Methods

<u>Method</u>	<u>Estimated Run Time</u> (arbitrary units)
MG (2nd order) (20x2x1.75)	70
MG (Fletcher-Powell) (20x3x1.75)	105
Second Variational (Mitter) (20x4x1)	80
Conjugate Gradient (4x30x1)	120

Again the results tend to favor the MG methods, although in this case the margin is perhaps not decisive.

To summarize this section, it should be pointed out that the problem discussed above has free terminal conditions, and is therefore unfavorable to the MG method. It lends itself especially well to solutions by control iteration methods, however, because the appropriate final adjoint values (namely zero) are known in advance. By contrast, the basic advantage of the MG methods does not apply to the open-terminal case because there is no way to reduce the dimension of the upper-level search.

From these results, it may be concluded that, even for an unfavorable example, the MG methods (a) have definitely better overall convergence rates than control iteration methods, and (b) range from slightly better to competitive on the basis of run time.

#### 4.3.2.2 Constrained Problems

It was originally stated that the MG methods are primarily intended for heavily constrained problems, so that even better performance should be expected in that case. The effect of various types and numbers of constraints will be considered next.

Effect of terminal and intermediate boundary conditions. - When imposed upon the preceding problem, these actually result in improved MG performance by limiting the dimensions of the upper level search. Point boundary values are incorporated especially readily in the MG/Fletcher-Powell formulation. The constrained coordinates are simply excluded from the search vector and the remaining ones are renumbered. Typical results are shown in Figure 4-7, where run time (measured in the same arbitrary but consistent units shown in Table 4-3) is plotted against the number of point constraints. For the MG method, the run time was 105 units (as in Table 4-3) for fixed initial point only, and decreased gradually as more and more constraints were added. Control iteration methods on the other hand show a rapid increase because the factor  $N_{bv}$  increases from 1 to at least 2 (possibly much more) in going to a 2 point boundary value problem. It is probable that even further increases would attend intermediate boundary values (in general, an extra set of multipliers is involved with each added point constraint) and there appears to be no likelihood of a compensatory decrease in  $N_{ds}$ . These results lead to the conclusion that even if the MG method is inferior for open

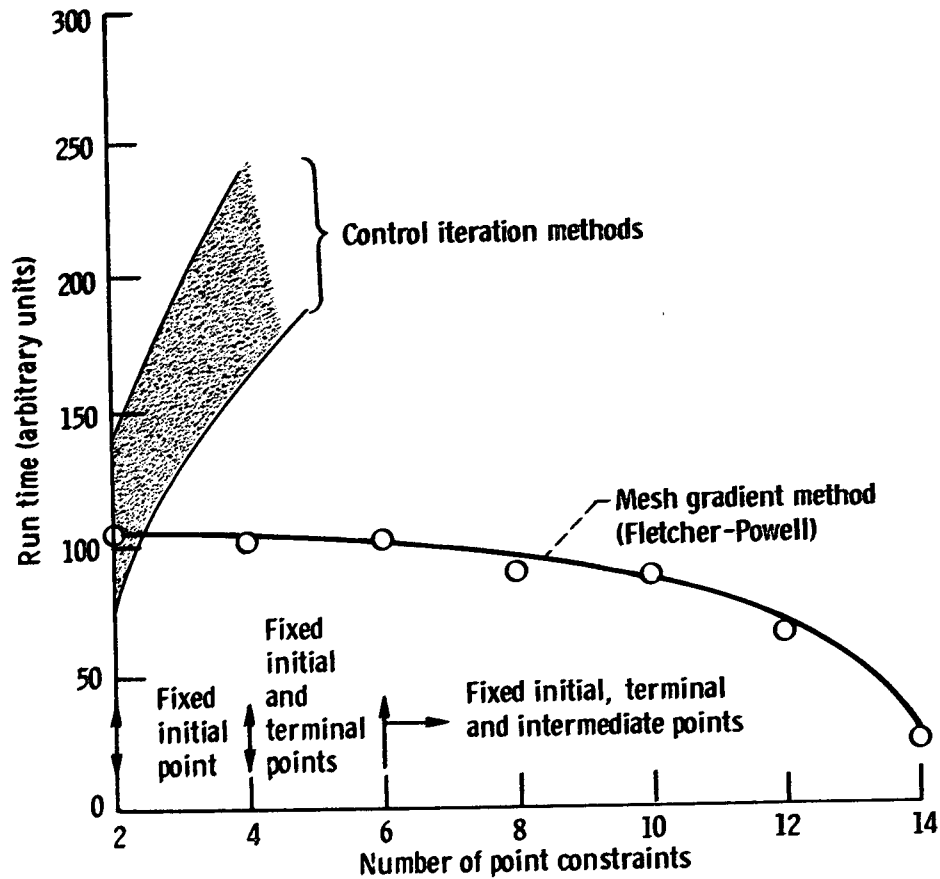


FIGURE 4-7. - EFFECT OF TERMINAL AND INTERMEDIATE BOUNDARY CONDITIONS. 0.5 PERCENT CONVERGENCE.

terminal problems, it will surpass the control iteration method eventually, if enough terminal boundary values (not to mention intermediate boundary values) are imposed.

Effect of state-variable point inequality constraints. - Another advantage of the programming formulation of the MG method is that state-variable inequality constraints can be incorporated point wise with no difficulty. A series of runs, similar to that discussed above except with equality constraints replaced by consistent inequalities, yielded the following result: In no case was the 105-unit run time of the original unconstrained case exceeded.

Note, that the point-wise inequality constraint is sufficient for many applications, either as an approximation or because it is already an adequate description of what is required.\*

#### 4.3.2.3 Effect of Poor Starting Iterates

The initial approximation used so far is quite simple, and is evidently no better than that used for the control iteration methods. Nevertheless, the possibility exists that both estimates may have been fortuitously good. Therefore, the previous initial estimate for the MG method was perturbed by adding to each partition point a random point in the interval  $\pm \frac{t}{50} (10-t)$ . Although results varied somewhat, as might be expected, no case of convergence failure was

---

\* For example, because of the hazard from solar flares, it may be required that the point of closest solar approach along an interplanetary trajectory shall not be less than (say) 0.4 Astronomical Units.

encountered. Generally, no more than 2 or 3 iterations were required to recover a case as good as the original exponential approximation yielded. The remaining steps then approach the original pattern (c.f., Fig. 4-6). A typical case is listed below.

Table 4-4

Iteration History From Poor Initial Trajectory  
(Typical Example)

<u>Iteration Number</u>	<u>Value of J</u>
0	108.9 (initial)
1	48.6
2	28.1
3	23.0
4	21.8
5	21.5 (0.5% convergence)

4.3.2.4 Effect of Varying the Distribution of Computational Effort

It may be noted that the distribution of effort between the levels is essentially arbitrary, provided the subarcs are short enough to avoid instabilities. That is, it is possible to set up the problem on the basis of a few subarcs each containing many integration steps, or vice-versa. Imagine that this distribution is parameterized by  $\beta$  (where  $0 \leq \beta \leq 1$ ), so defined that  $\beta = 0$  means the entire burden is carried by the upper level and  $\beta = 1$  means that the lower level carries the entire load. For the present discussion  $\beta$  is arbitrarily defined by the relation

$$\beta = 1 - \frac{N_{is}}{N_{sa}} \quad (6)$$

where

$N_{sa}$  = integer number of subarcs

and

$N_{is}$  = integer number of integration steps

When  $\beta = 0$ , a fairly conventional initial-value method is obtained as a special case of the MG approach. The special case obtained at the other extreme ( $\beta = 1$ ) is basically equivalent to the generalized Newton-Raphson algorithm. The best computational results, however, are obtained for intermediate values of  $\beta$ . This is illustrated in Figure 4-8, where run time (normalized so that the minimum value is unity) is plotted against  $\beta$  for the original example (free terminal problem). A value of  $N_{is} = 36$  was used for all cases, the steps being distributed in an appropriate manner between the subarcs. Clearly there is a very broad minimum of run time around  $\beta = 0.14$  to  $0.20$  for this problem ( $N_{sa} = 6$  to  $8$ ). For low values of  $\beta$ , the bottom level boundary value iterations are seriously hampered by the previously described numerical instabilities. For increasing  $\beta$ , these boundary value searches improve rapidly and soon reach a point where, for all practical purposes, 1-step convergence is obtained. Further increases of  $\beta$  cannot improve the lower-level operation beyond this point, but does increase the dimension of the upper level search. Thus, the run time gradually increases as  $\beta \rightarrow 1$ .

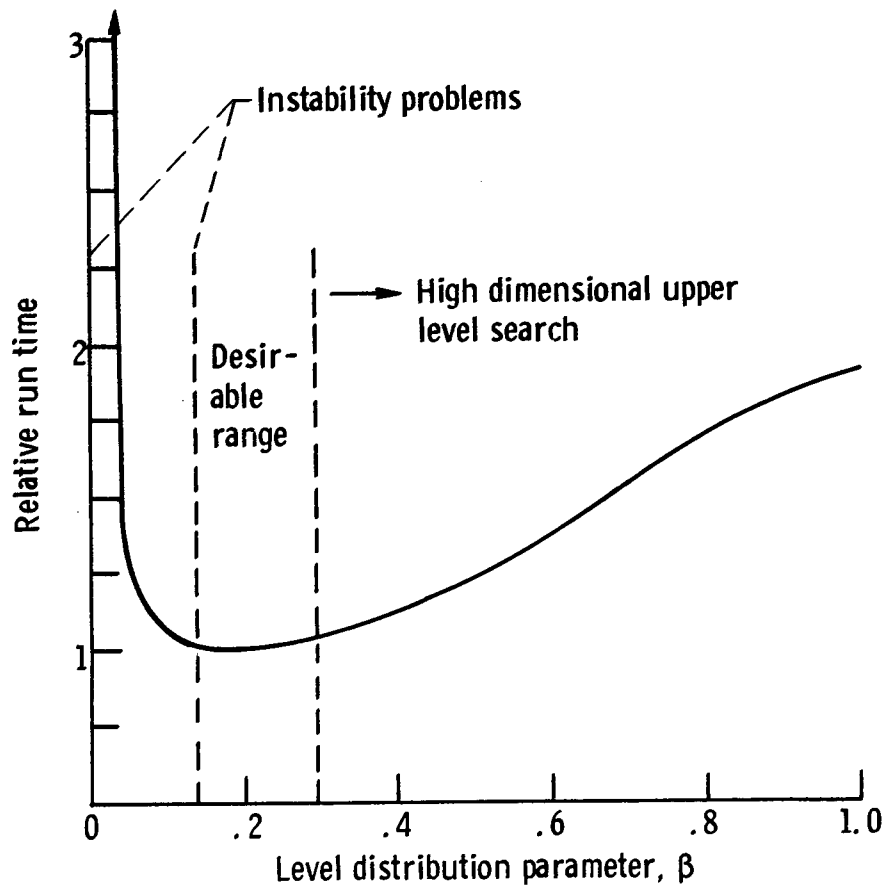


FIGURE 4-8. - EFFECT OF VARYING THE DISTRIBUTION OF COMPUTATIONAL EFFORT BETWEEN LEVELS.

There are two important points to be noted, from this example. First, the best results are found for the general case, and not for either of the special cases which existed previously. Secondly, the importance of choosing the appropriate value of  $\beta$  has been demonstrated. In fact, the example suggests the possibility of devising an adaptive loop (in effect, a third "level" of the Mesh Gradient method) which would vary  $\beta$  in an efficient if not strictly optimal fashion, as the problem runs. This is considered to be a valuable potential feature of the present approach and warrants further study and experimentation.

#### 4.4 Example 4 - Optimal Space Trajectories

As a final example, the Mesh Gradient technique is applied to the following rendezvous problem in space trajectory mechanics (see Fig. 4-9).

$$\text{Minimize } J = \frac{1}{2\beta} \int_{t_0}^{t_K} |\vec{a}(t)|^2 dt + \frac{\beta}{2} \sum_{k=0}^K |\vec{v}_k|^2 \quad (1)$$

with

$$\dot{\vec{x}}(t) = \vec{v}(t) \quad \text{and} \quad \dot{\vec{v}}(t) = -\omega^2(\vec{x})\vec{x}(t) + \vec{a}(t) \quad (2)$$

where

$$\omega^2 = [x^2(t) + y^2(t) + z^2(t)]^{-3} \quad (3)$$

$x$ ,  $y$ , and  $z$  are the three components of  $\vec{x}$  in  $R_3$ ;  $t_0$  and  $t_K$  are fixed; the initial and final boundary conditions are

$$\begin{aligned} \vec{x}(t_0) &= \vec{x}_0 & \vec{v}(t_0) &= \vec{v}_0 \\ \vec{x}(t_K) &= \vec{x}_K & \vec{v}(t_K) &= \vec{v}_K \end{aligned}$$

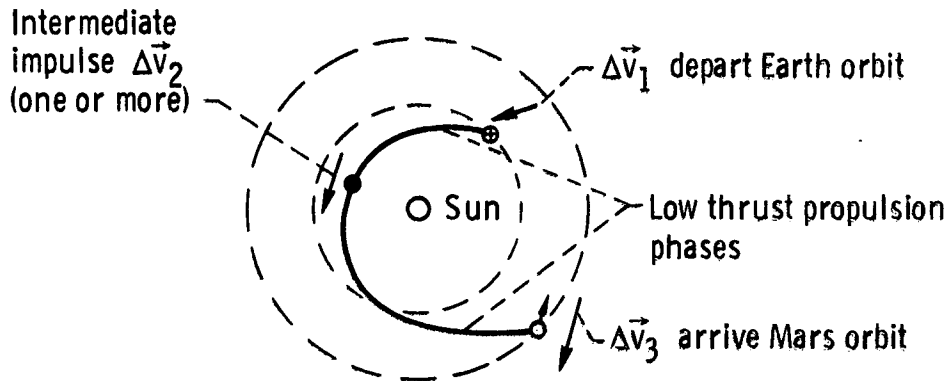


FIGURE 4-9. - TYPICAL MULTI-BURN SPACE RENDEZVOUS TRAJECTORY.

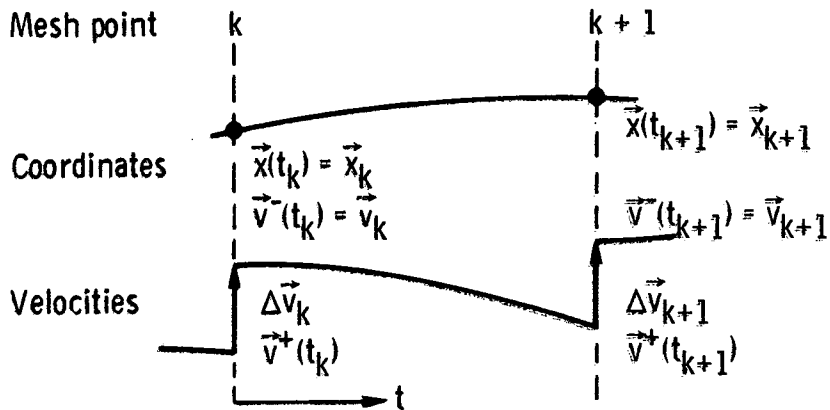


FIGURE 4-10. - BOUNDARY CONDITIONS FOR HIGH AND LOW THRUST SUBARC.

(Additional constraints may apply at the intermediate times  $t_1 \dots t_{K-1}$ .) Physically, this system approximates a rocket having both high and low-thrust propulsion devices. See Figure 4-10.

The high thrust propulsion phases are represented, as usual, by impulsive velocity changes at  $K + 1$  discrete points along the trajectory. Each of these is considered to be accomplished in a single step, such that the actual velocity resulting at the end of one arc,  $\vec{v}^+(t_k)$  is instantaneously corrected to equal the nominal velocity  $\vec{v}_k$  defined for the  $k^{\text{th}}$  mesh point. Thus,

$$\Delta \vec{v}_k = \vec{v}^+(t_k) - \vec{v}_k \quad (5a)$$

(The initial velocity for the succeeding arc,  $\vec{v}^-(t_k)$ , is set equal to the nominal velocity  $\vec{v}_k$ , i.e.,  $\vec{v}^-(t_k) = \vec{v}_k$ . The cost is computed as

$$|\Delta \vec{v}_k|^2 = |\vec{v}^+(t_k) - \vec{v}_k|^2 \quad (5b)$$

Note, that the total point-value contribution to  $J$  thus reflects a sum of squares, as opposed to the summation of magnitudes usually considered. This choice emphasizes the effect of large individual  $\Delta \vec{v}$ 's and is hence considered more appropriate for the case where an individual rocket stage is to be used for each high-thrust maneuver.

The cost of the low thrust propulsion phase is represented by the integral term in Eq. (1). The integral involves  $|\vec{a}|^2$  rather than  $|\vec{a}|$ ; thus, we are considering a variable rather than constant thrust device. This choice was dictated by convenience; however, it is known that variable-thrust performance can be well approximated

by as little as 2 or 3 optimal constant thrust levels (ref. 46). In addition, methods exist for deriving constant-thrust performance from variable-thrust results (ref. 77 and 186).

Finally, the parameter  $\beta$  indicates the relative efficiency of the low-thrust and high-thrust systems. The trajectory solutions will vary from all-impulsive at  $\beta = 0$  to 100% low variable thrust as  $\beta \rightarrow \infty$ . For intermediate values of  $\beta$ , the solution is required to indicate the optimum split between high- and low-thrust propulsion.

#### 4.4.1 Subarc Solutions

A solution by Picard iteration may be obtained by assuming that  $\omega^2$  in Eq. (2) is a known function of time (initially approximated by the method of Appendix C), solving the resulting linear-quadratic problems, computing the new  $\omega^2$  function by Eq. (3), and iterating to convergence.

When the maximum principle is applied, we obtain (for each member of the sequence) the following canonical system.

$$\begin{aligned}\dot{\vec{x}} &= \vec{v} \\ \dot{\vec{v}} &= -\omega^2(t)\vec{x} + \beta\vec{\mu} \\ \dot{\vec{\lambda}} &= \omega^2(t)\vec{\mu} \\ \dot{\vec{\mu}} &= -\vec{\lambda}\end{aligned}\tag{6}$$

where  $\vec{\lambda}$  and  $\vec{\mu}$  are the adjoint vectors corresponding to  $\vec{x}$  and  $\vec{v}$ , respectively.

#### 4.4.1.1 Boundary Conditions

As indicated in Figure 4-10, the applicable boundary and transversality conditions, for fixed end points  $\vec{x}_k$  and  $\vec{x}_{k+1}$ , are

$$\begin{aligned}\vec{x}(t_k) - \vec{x}_k &= 0 \\ \vec{v}^-(t_k) - \vec{v}_k &= 0 \\ \vec{x}(t_{k+1}) - \vec{x}_{k+1} &= 0 \\ -\vec{\mu}(t_{k+1}) + \frac{\partial}{\partial \vec{v}^+(t_{k+1})} \frac{\beta}{2} \sum_{k=0}^K |\vec{\nabla} \vec{v}|_k^2 &= 0\end{aligned}\tag{7a}$$

Or, using (5),

$$\begin{aligned}\vec{x}(t_k) - \vec{x}_k &= 0 \\ \vec{v}^-(t_k) - \vec{v}_k &= 0 \\ \vec{x}(t_{k+1}) - \vec{x}_{k+1} &= 0 \\ -\vec{\mu}(t_{k+1}) + \beta(\vec{v}^+(t_{k+1}) - \vec{v}_{k+1}) &= 0\end{aligned}\tag{7b}$$

If on the other hand the end point  $\vec{x}_{k+1}$  as well as the nominal velocity  $\vec{v}_{k+1}$  is considered to be variable, the applicable boundary and transversality conditions are

$$\begin{aligned}\vec{x}(t_k) - \vec{x}_k &= 0 \\ \vec{v}^-(t_k) - \vec{v}_k &= 0 \\ -\vec{\lambda}^+(t_{k+1}) + \frac{\partial}{\partial \vec{x}_{k+1}} \frac{\beta}{2} \sum_{k=0}^K |\Delta \vec{v}|_k^2 &= 0 \\ -\vec{\mu}^+(t_{k+1}) + \frac{\partial}{\partial \vec{v}^+(t_{k+1})} \frac{\beta}{2} \sum_{k=0}^K |\Delta \vec{v}|_k^2 &= 0\end{aligned}$$

Again using Eq. (5) and the chain rule for partials with respect to  $\vec{x}$ , these may be written

$$\vec{\lambda}^-(t_k) + \beta \left[ \Delta \vec{v}_k \left[ \frac{\partial \Delta \vec{v}_k}{\partial \vec{x}_k} \right] + \Delta \vec{v}_{k+1} \left[ \frac{\partial \Delta \vec{v}_{k+1}}{\partial \vec{x}_k} \right] \right] = 0$$

$$\vec{\mu}^-(t_k) + \beta \Delta \vec{v}_k = 0$$

$$-\vec{\lambda}^-(t_{k+1}) + \beta \left\{ \Delta \vec{v}_{k+1} \left[ \frac{\partial \Delta \vec{v}_{k+1}}{\partial \vec{x}_{k+1}} \right] + \Delta \vec{v}_k \left[ \frac{\partial \Delta \vec{v}_k}{\partial \vec{x}_{k+1}} \right] \right\} = 0$$

$$-\vec{\mu}^-(t_{k+1}) + \beta \Delta \vec{v}_{k+1} = 0$$

In applying Eqs. (8b), it is important to note that a variation in  $\vec{x}_k$  may, in general, change the value of  $|\Delta \vec{v}|_{k-1}^2$  and  $|\Delta \vec{v}|_{k+1}^2$  as well as  $|\Delta \vec{v}|_k^2$ . (Recall Eq. 3.2.2(11).)

The solution of (6) to (8) may be simplified considerably by noting that the three (x,y, and z) components of the solution are independent and may be computed separately. That is, a 4x4 fundamental matrix is computed and then applied to each of the 3 components, as opposed to a single operation involving a 12x12 matrix.

#### 4.4.1.2 General-Case and Low Thrust Solutions in Terms of the Fundamental Matrix

Let us write the 4x4 fundamental matrix as

$$\Phi(t; t_0) = \begin{bmatrix} a_{11} & a_{12} & | & b_{11} & b_{12} \\ a_{21} & a_{22} & | & b_{21} & b_{22} \\ c_{11} & c_{12} & | & d_{11} & d_{12} \\ c_{21} & c_{22} & | & d_{21} & d_{22} \end{bmatrix} \quad (9)$$

where

$$\Phi(t; t_0) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega^2 & 0 & 0 & \beta \\ 0 & 0 & 0 & \omega^2 \\ 0 & 0 & -1 & 0 \end{bmatrix}_t \Phi(t; t_0) \quad (10)$$

and

$$\Phi(t_0; t_0) = I$$

so that

$$\begin{bmatrix} x \\ v \\ \lambda \\ \mu \end{bmatrix}_{t_{k+1}} = \Phi(t_{k+1}; t_k) \begin{bmatrix} x \\ v \\ \lambda \\ \mu \end{bmatrix}_{t_k} \quad (11)$$

Combining (7), (9), and (11) we see that

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ a_{11} & a_{12} & b_{11} & b_{12} \\ \beta a_{21} - c_{21} & \beta a_{22} - c_{22} & \beta b_{21} - d_{21} & \beta b_{22} - d_{22} \end{bmatrix} \begin{bmatrix} x \\ v \\ \lambda \\ \mu \end{bmatrix}_{t_k} \quad (12)$$

$$= \begin{bmatrix} x_k \\ v_k \\ x_{k+1} \\ \beta v_{k+1} \end{bmatrix}$$

Or, after solving analytically for

$$x(t_k) = x_k \quad (13)$$

and

$$\bar{v}^-(t_k) = v_k$$

the unknown adjoint variables at the beginning of the arc are given by

$$\begin{bmatrix} b_{11} & | & b_{12} - 1/\beta \\ \beta b_{21} - d_{21} & | & \beta b_{22} - d_{22} - \frac{1}{\beta} \end{bmatrix} \begin{bmatrix} \lambda^- \\ \mu^- \end{bmatrix}_{t_k} = \begin{bmatrix} x_{k+1} - a_{11}x_k - a_{12}v_k \\ \beta v_{k+1} - (\beta a_{21} + c_{21})x_k - (\beta a_{22} + c_{22})v_k \end{bmatrix} \quad (14)$$

Inverting (14) we have

$$\begin{bmatrix} \lambda^- \\ \mu^- \end{bmatrix}_{t_k} = \frac{1}{\Delta} \begin{bmatrix} \beta b_{22} - d_{22} - 1/\beta & | & 1/\beta - b_{12} \\ -\beta b_{21} d_{21} & | & b_{11} \end{bmatrix} \begin{bmatrix} x_{k+1} - a_{11}x_k - a_{12}v_k \\ \beta v_{k+1} - (\beta a_{21} + c_{21})x_k - (\beta a_{22} + c_{22})v_k \end{bmatrix} \quad (15)$$

where the determinant is

$$\Delta = b_{11}(\beta b_{22} - d_{22} - 1/\beta) + (b_{12} - 1/\beta)(-\beta b_{21} + d_{21})$$

When  $\beta \rightarrow \infty$ , it can be shown that  $\lambda^-(t_k)$  and  $\mu^-(t_k)$  reduce to those values that result from the standard, low-variable-thrust rendezvous problem, i.e.,

$$\begin{aligned} \mu^-(t_k) &= \frac{-b_{11}}{b_{11}b_{22} - b_{12}b_{21}} (v_{k+1} - a_{21}x_k - a_{22}v_k) \\ &\quad + \frac{b_{21}}{b_{11}b_{22} - b_{12}b_{21}} (x_{k+1} - a_{11}x_k - a_{12}v_k) \\ \lambda^-(t_k) &= \frac{b_{22}}{b_{11}b_{22} - b_{12}b_{21}} (x_{k+1} - a_{11}x_k - a_{12}v_k) \\ &\quad - \frac{b_{12}}{b_{11}b_{22} - b_{12}b_{21}} (v_{k+1} - a_{21}x_k - a_{22}v_k) \end{aligned} \quad (16)$$

#### 4.4.1.3 Solutions for the Case of Impulsive Thrust

When  $\beta \rightarrow 0$  in the impulsive-thrust limit, both  $\vec{\lambda}(t)$  and  $\vec{\mu}(t)$  vanish over the entire interval, and no low thrust propulsion is used. In this case the initial and final velocities,  $\vec{v}^-(t_k)$  and  $\vec{v}^+(t_{k+1})$  must be chosen so that the desired mesh coordinates  $\vec{x}_k$  and  $\vec{x}_{k+1}$  are achieved at the end points of the trajectory. That is, when the applicable upper-left block of  $\Phi$  in Eq. (9) is re-labeled as  $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ , we have

$$\begin{bmatrix} \vec{x} \\ \vec{v}^+ \end{bmatrix}_{t_{k+1}} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{v}^- \end{bmatrix}_{t_k} \quad (17)$$

from which we find that the required initial velocity at  $t_k$  is

$$\vec{v}^-(t_k) = b^{-1}(\vec{x}_{k+1} - a\vec{x}_k) \quad (18)$$

and the corresponding (dependent) final velocity at  $t_{k+1}$  is

$$\vec{v}^+(t_{k+1}) = (c - db^{-1}a)\vec{x}_k - b^{-1}a\vec{x}_{k+1} \quad (19)$$

#### 4.4.2 Mesh-Point Iteration

When the preceding results are substituted into the general equation for the gradient vector at a general ( $k^{\text{th}}$ ) mesh point, we obtain

$$\nabla_k J = \frac{\begin{bmatrix} \vec{\lambda}^+(t_k) - \vec{\lambda}^-(t_k) + \beta \sum_{j=k-1}^{k+1} \Delta \vec{v}_j \left[ \frac{\partial \Delta \vec{v}_j}{\partial \vec{x}_k} \right] \\ \vec{\mu}^+(t_k) - \vec{\mu}^-(t_k) + \beta \sum_{j=k}^{k+1} \Delta \vec{v}_j \left[ \frac{\partial \Delta \vec{v}_j}{\partial \vec{v}_k} \right] \end{bmatrix}}{\quad} \quad (20)$$

for the general, high + low thrust case. Notice that a variation of a mesh point coordinate  $\vec{x}_k$  will be general change the values of  $|\Delta\vec{v}|_{k-1}^2$  and  $|\Delta\vec{v}|_{k+1}^2$  as well as  $|\Delta\vec{v}|_k^2$ . But since a velocity-discontinuity is permitted, a change in the desired velocity  $\vec{v}_k$  at a mesh point will affect the values of  $|\Delta\vec{v}|_k^2$  and  $|\Delta\vec{v}|_{k+1}^2$ , but not  $|\Delta\vec{v}|_{k-1}^2$ .

For the low thrust system, the above reduces to

$$\nabla_k^J = \begin{bmatrix} \vec{\lambda}^+(t_k) - \vec{\lambda}^-(t_k) \\ \vec{\mu}^+(t_k) - \vec{\mu}^-(t_k) \end{bmatrix} = \begin{bmatrix} \Delta\vec{\lambda} \\ \Delta\vec{\mu} \end{bmatrix}_k \quad (21)$$

For the impulsive case, the mesh-point initial velocities are no longer independent but must be chosen so that the prescribed coordinates are attained at the end points of each subarc. Thus, the gradient is with respect to coordinates only, i.e.,

$$\nabla_k^J = \left[ \sum_{j=k-1}^{k+1} \Delta\vec{v}_j \left[ \frac{\partial \Delta\vec{v}_j}{\partial \vec{x}_k} \right] \right] \quad (22)$$

As before, the gradient in  $\vec{x}_{pp}$  may be constructed by simply adjoining the column vectors (20, (21), or (22), taking care that the summation index  $j$  remains in the same range ( $0 \leq j \leq K$ ) as does the mesh point index  $k$ .

The required partial derivatives may be readily computed from Eq. (19) to be

$$\frac{\partial \Delta\vec{v}_j}{\partial \vec{x}_k} = \sum_{j=k-1}^{k+1} \left\{ \frac{\partial \vec{v}_j}{\partial \vec{x}_k} - \frac{\partial \vec{v}_j}{\partial \vec{x}_k} \right\} \quad (23)$$

where

$$\frac{\partial \Delta \vec{v}_{k-1}}{\partial \vec{x}_k} = - \frac{\partial \vec{v}_{k-1}^-}{\partial \vec{x}_k} = - b^{-1} \Big|_{k-1}$$

$$\frac{\partial \Delta \vec{v}_{k+1}}{\partial \vec{x}_k} = \frac{\partial \vec{v}_{k+1}^+}{\partial \vec{x}_k} = (c - db^{-1}a)_k$$

and

$$\frac{\partial \Delta \vec{v}_k}{\partial \vec{x}_k} = \frac{\partial \vec{v}_k^+}{\partial \vec{x}_k} - \frac{\partial \vec{v}_k^-}{\partial \vec{x}_k} = db^{-1} \Big|_{k-1} + b^{-1}a \Big|_k$$

Thus, in  $\mathbf{X}_{pp}$  we have

$$\nabla_{\mathbf{X}} \mathbf{J} = \begin{bmatrix} (B^{-1}A)_0 \Delta \vec{v}_0 + (C - DB^{-1}A)_0 \Delta \vec{v}_1 + 0 \\ -B_0^{-1} \Delta \vec{v}_0 + \left[ (DB^{-1})_0 + (B^{-1}A)_1 \right] \Delta \vec{v}_1 + (C - DB^{-1}A)_1 \Delta \vec{v}_2 \\ \vdots \\ \vdots \\ -B_{K-2}^{-1} \Delta \vec{v}_{K-2} + \left[ (DB^{-1})_{K-2} + (B^{-1}A)_{K-1} \right] \Delta \vec{v}_{K-1} + (C - DB^{-1}A)_{K-1} \Delta \vec{v}_K \\ -B_{K-1}^{-1} \Delta \vec{v}_{K-1} + \left[ (DB^{-1})_{K-1} + (B^{-1}A)_K \right] \Delta \vec{v}_K + 0 \end{bmatrix} \quad (24)$$

The partial derivatives for the general case, Eq. (20), may be computed from the relations

$$\left. \begin{aligned} \frac{\partial \Delta \vec{v}_k}{\partial \vec{v}_k} &= \frac{\vec{v}_k^+}{\partial \vec{v}_k} - \mathbf{I} \\ \frac{\partial \Delta \vec{v}_{k+1}}{\partial \vec{v}_k} &= \frac{\vec{v}_{k+1}^+}{\partial \vec{v}_k} \\ \frac{\partial \Delta \vec{v}_k}{\partial \vec{x}_k} &= \frac{\vec{v}_k^+}{\partial \vec{x}_k} \\ \frac{\partial \Delta \vec{v}_{k+1}}{\partial \vec{x}_k} &= \frac{\vec{v}_{k+1}^+}{\partial \vec{x}_k} \end{aligned} \right\} \quad (25)$$

where

$$\left. \begin{aligned} \frac{\partial \vec{v}_k^+}{\partial \vec{v}_k} &= b_{21} \Big|_{k-1} \frac{\partial \vec{\lambda}_{k-1}^-}{\partial \vec{v}_k} + b_{22} \Big|_{k-1} \frac{\partial \vec{\mu}_{k-1}^-}{\partial \vec{v}_k} \\ \frac{\partial \vec{v}_k^+}{\partial \vec{x}_k} &= b_{21} \Big|_{k-1} \frac{\partial \vec{\lambda}_{k-1}^-}{\partial \vec{x}_k} + b_{22} \Big|_{k-1} \frac{\partial \vec{\mu}_{k-1}^-}{\partial \vec{x}_k} \\ \frac{\partial \vec{v}_{k+1}^+}{\partial \vec{x}_k} &= b_{21} \Big|_k \frac{\partial \vec{\lambda}_k^-}{\partial \vec{x}_k} + b_{22} \Big|_k \frac{\partial \vec{\mu}_k^-}{\partial \vec{x}_k} \\ \frac{\partial \vec{v}_{k+1}^+}{\partial \vec{v}_k} &= b_{21} \Big|_k \frac{\partial \vec{\lambda}_k^-}{\partial \vec{v}_k} + b_{22} \Big|_k \frac{\partial \vec{\mu}_k^-}{\partial \vec{v}_k} \end{aligned} \right\} \quad (26)$$

The eight partials of  $\vec{\mu}^-$  and  $\vec{\lambda}^-$  in Eq. (26) are in turn computed by differentiating Eq. (15) and re-labeling indices. For example,

$$\frac{\partial \vec{\lambda}_k}{\partial \vec{x}_k} = \left[ \begin{array}{l} - \frac{(\beta b_{22} - d_{22} - 1/\beta) a_{11}}{b_{11}(\beta b_{22} - d_{22} - 1/\beta) + (b_{12} - 1/\beta)(-\beta b_{21} + d_{21})} \\ + \frac{(1/\beta - b_{12})(\beta a_{21} + c_{21})}{b_{11}(\beta b_{22} - d_{22} - 1/\beta) + (b_{12} - 1/\beta)(-\beta b_{21} + d_{21})} \end{array} \right] \quad (27)$$

and the other seven are similarly defined.

The gradients computed as above are finally used to drive a Fletcher-Powell minimization routine, c.f., Appendix B.

#### 4.4.3 Numerical Results

##### 4.4.3.1 Accuracy of the Approximate Solution

As point out in Section 4.4.1, solutions may be computed by the method of Picard iteration. On the other hand, Appendix C shows an approximate analytical form for the  $\omega$  function which is "exact" at two points of the trajectory and permits a closed form solution for the state and adjoint variables. By inspecting the defining Eqs. 10(4) and 10(5) for  $\omega$ , it may be seen that  $\dot{\omega}$  never changes sign and  $\omega$  itself is monotonic. By contrast, the "real"  $\omega$  function (Eq. 4.4(3)) has  $\dot{\omega} = 0$  at every apse or turning-point of the trajectory and cannot be well-approximated by a monotonic function. On the other hand, the "real" function can evidently be approximated as well as we please by adjoining a number of monotonic segments, each having its own appropriately chosen value of  $K$  in Eq. 10(5). This is illustrated in Figure 4-11 where  $\omega$  functions are plotted against heliocentric travel angle for an Earth- to Mars-transfer trajectory such as the one illustrated previously in Figure 4-9. The solid

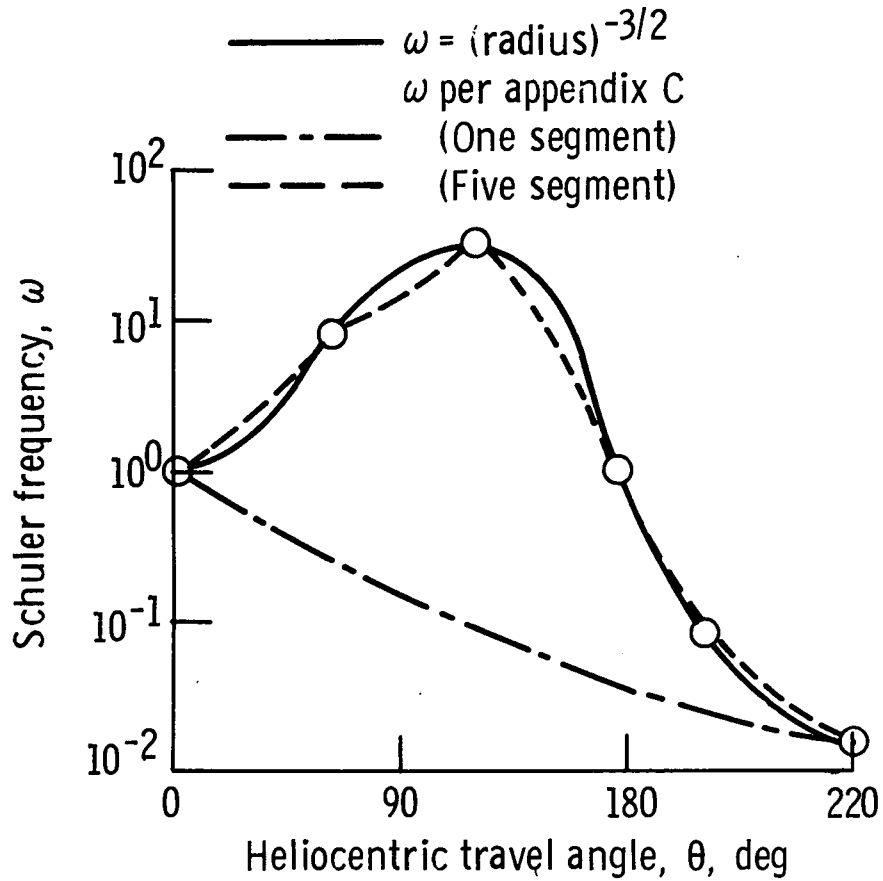


FIGURE 4-11. - SCHULER FREQUENCIES ( $\omega$ ) ALONG  $300^\circ$ , 250-DAY EARTH-TO MARS ORBIT RENDEZVOUS TRAJECTORY.

curve is the "real" function, the dashed curve shows the single segment approximation - which as expected is extremely poor - and the dotted curve shows a five-segment approximation closely approaching the "real" result.

Clearly, this approach is compatible with the Mesh Gradient technique, with  $\omega$  being approximated by Eq. 10(5) on each of the Level I subarcs. The descent process is illustrated in Figure 4-12, for a low-variable-thrust vehicle flying the same Earth-to-Mars orbit rendezvous mission considered above. The current criterion value,  $J_n$  is plotted against the iteration number  $n$ , and compared with the "exact", numerically integrated result obtained from reference 112. The case  $n = 0$  represents a first or preliminary guess using only one trajectory arc; as could be expected, it is a poor approximation. Nevertheless, when coordinates and velocities from the single arc approximation were used to define mesh points for a 5-arc approximation, the much-improved value shown at  $n = 1$  resulted. A Fletcher-Powell search (initialized by the  $n = 1$  trajectory) then resulted in the descent sequence denoted by  $n = 2, 3, 4, 5, \dots$ ; with convergence obtained, for all practical purposes, by the fifth step.

The effect of  $K$ , the number of mesh-points used, is illustrated in Figure 4-13. Here the value of  $J$  obtained at the end of the descent process is shown as a function of  $K$  (or the number of subarcs,  $= K-1$ ). Clearly, the original error is drastically reduced by using even 2 subarcs, and 5 subarcs yield a rather accurate approximation.

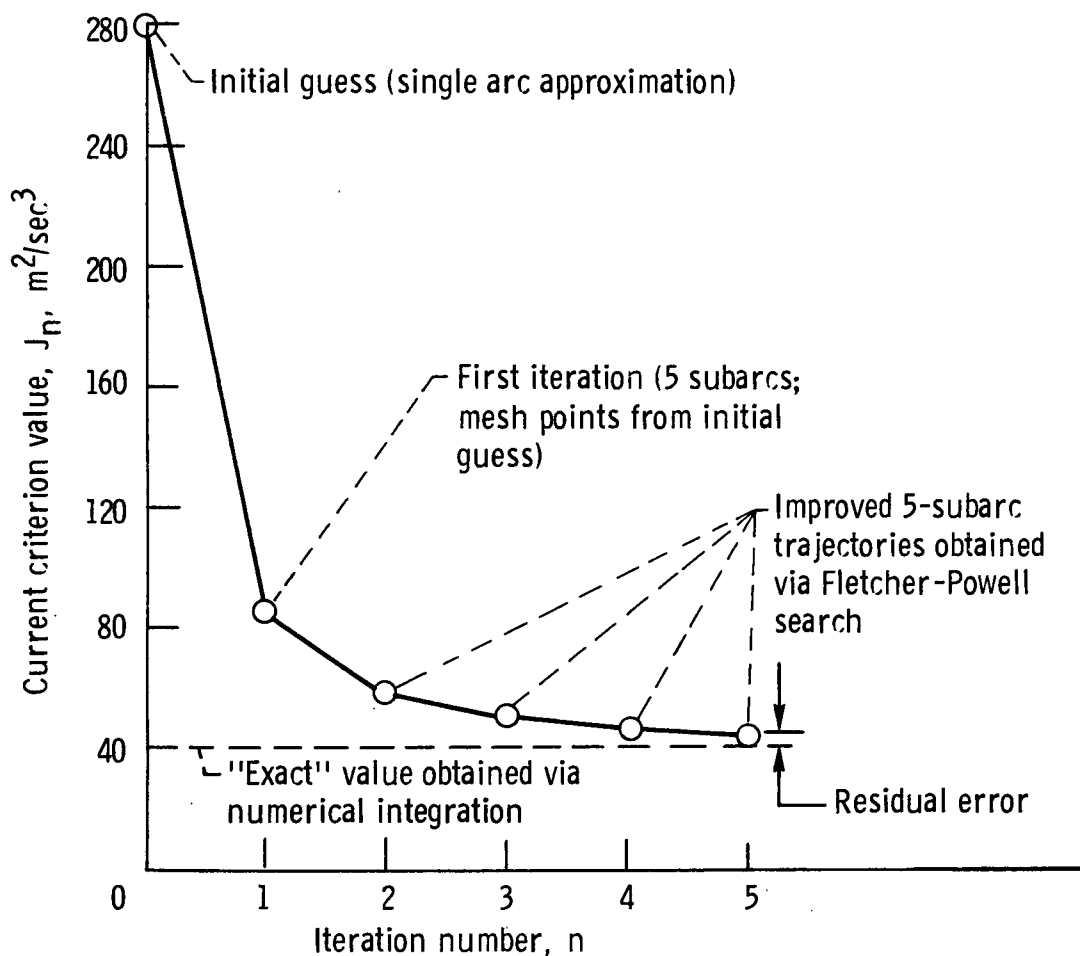


FIGURE 4-12. - CRITERION VALUE VS ITERATION NUMBER FOR LOW-VARIABLE THRUST,  $300^\circ$ , 250 DAY EARTH-MARS ORBIT RENDEZVOUS TRAJECTORY.

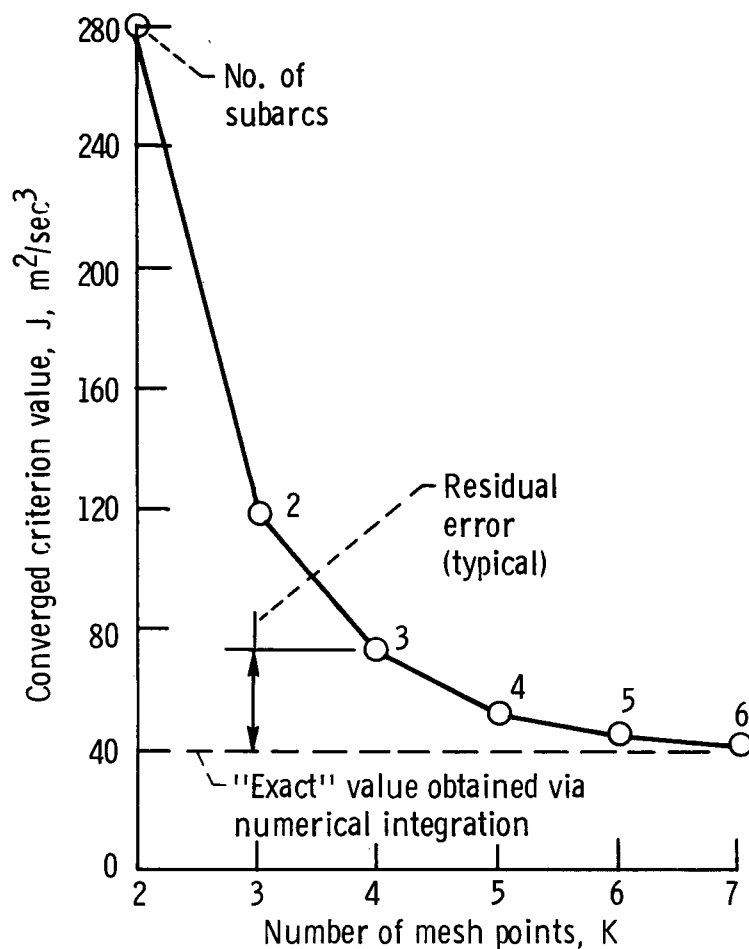


FIGURE 4-13. - EFFECT OF NUMBER OF MESH POINTS ON THE CRITERION VALUE OBTAINED. LOW-VARIABLE THRUST,  $300^\circ$ , 250-DAY EARTH-MARS ORBIT RENDEZVOUS TRAJECTORY.

This illustrates the Mesh-Gradient method's ability to extend the validity of a useful but limited-range approximate solution. In the present example, this results in an estimated saving in computer time on the order of 10/1 compared to "exact" numerical-integration methods such as reference 112. (This is based upon a comparison of the number of computational steps required for each method; the reference computer program is no longer available for direct comparisons.)

#### 4.4.3.2 Low Variable Thrust Solutions

The above described procedure, using 5 subarcs, was applied to a range of Earth-orbit to Mars-orbit rendezvous trajectories. Results are shown in Figure 4-14 with  $J$  plotted against heliocentric travel angle  $\theta$  for a travel time of 250 days. The dotted portions of the curves denote cases where the vehicle makes less than one revolution about the sun; these are referred to as "direct" trajectories and are of course repeated every 360 degrees. The solid curve, however, represents "indirect" trajectories which wind fully around the sun before proceeding to the designated rendezvous point; these require more than one revolution.

Such "indirect", multi-revolution trajectories have proven troublesome in the past (computationally) due to numerical instabilities. This is because the multi-revolution trajectories generally involve close perihelion passages, so that the true  $\omega^2$  function is very large, for a time. This in turn causes the systems "effective" time constant to be short enough compared to typical travel times,

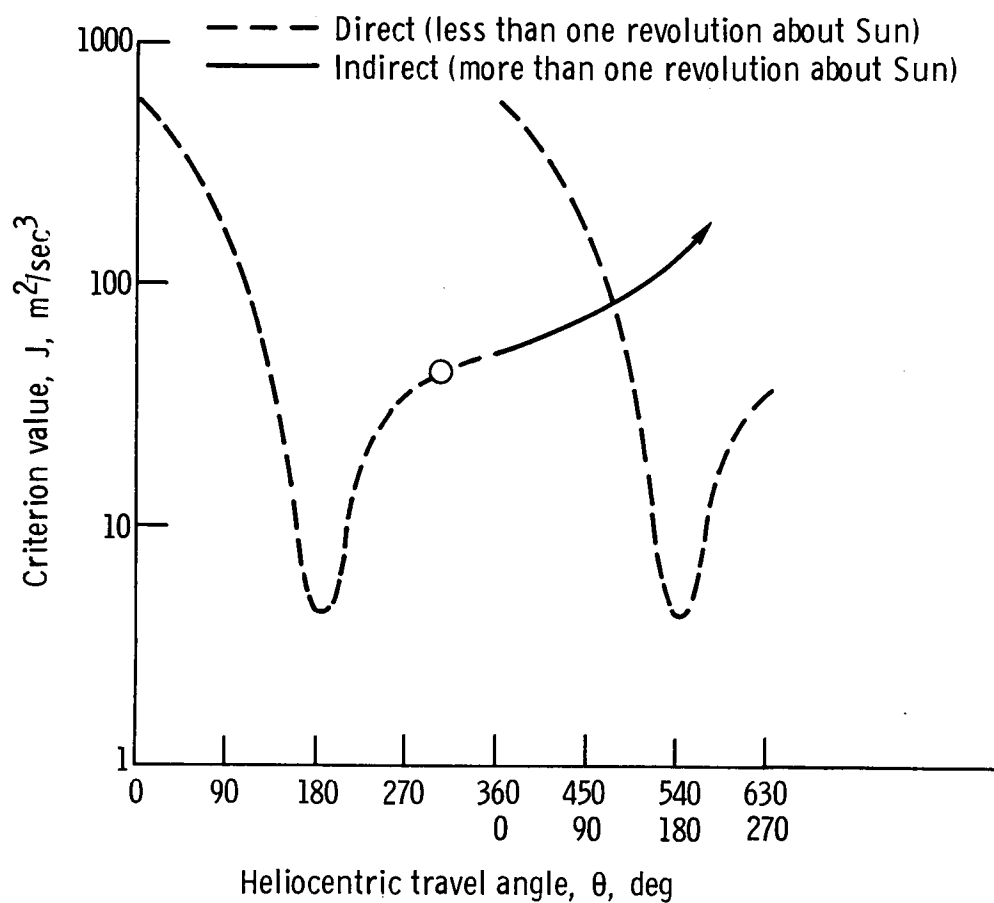


FIGURE 4-14. - COMPARISON OF DIRECT AND INDIRECT TRAJECTORIES, LOW VARIABLE THRUST, 250-DAY EARTH-MARS ORBIT RENDEZVOUS TRAJECTORIES. (CIRCULAR, COPLANAR PLANET ORBIT.)

that the results are extremely sensitive to small errors in initial estimates. For example, even the 300<sup>0</sup> example trajectory discussed above (noted on Fig. 4-14 by an asterisk) has a minimum solar distance of about 0.1 AU (Astronomical Unit), and stays below 0.25 AU for about 25 days out of the 250 day trip. During that time, the local "time constant" varies from 3 days to as little as 12 hours, and small perturbations may become grossly magnified.

Therefore, many previous trajectory computation methods have experienced significant difficulties in solving the multi-revolution TPBVP. Reference 112, for example, did not attempt to compute trajectory data beyond 330<sup>0</sup> even though it was intended as a major low-variable thrust trajectory data compilation.

#### 4.4.3.3 Impulsive Thrust Solutions

At the opposite extreme, we may consider the case where only the point-value contributions,  $\sum_0^K |\Delta \vec{v}|_k^2$  are significant. Then, as pointed out in Section 4.4.1.1, the low-thrust system is not used at all and Eqs. 4.4(6) may be written simply as

$$\begin{aligned}\ddot{\vec{x}} &= -\omega^2 \vec{x} \\ \ddot{\vec{\lambda}} &= \omega^2 \vec{\lambda}\end{aligned}\tag{28}$$

The appropriate fundamental matrix for Eq. (28) is also given in Appendix C, and the elements of the gradient vector were defined in Section 4.4.2. The resulting algorithm was then applied, as in the low variable thrust case, to a range of interplanetary transfers. Five subarcs were used for the sake of accuracy. Thus, up to six

impulses may be used. Descent and convergence properties were found to be substantially similar to those illustrated above in Figures 4-12 and 4-13, and will not be further discussed.

Figure 4-15 shows results obtained for a family of 250 day Earth orbit to Mars orbit rendezvous trajectories. The criterion is shown as a function of travel angle as before; note, however, that

$\sum_{k=0}^K |\Delta \vec{v}|_k$  rather than  $\frac{1}{2} \sum_{k=0}^K |\Delta v|_k^2$  is plotted. This is merely to

facilitate comparison with previous studies, such as reference 178; up to  $\theta = 360^\circ$  (the largest value considered in ref. 178) the results are indistinguishable. Both reference 178 and other available methods (e.g., Refs. 63-65, 73, and 74) could in principle be extended to operate beyond  $\theta = 360^\circ$ , but for practical reasons this has not been done and results comparable to the present ones are not available. It would appear, however, that the MG approach may have some significant computing-time advantage over the previous methods since (even for equal rates of convergence) the  $\omega$ -approximate sub-arc solutions require a smaller amount of calculations. In any case its time advantage over the reference 178 technique, the only one for which direct comparison are available, varies from 5/1 to 25/1 for computing a single trajectory. These numbers, which basically reflect the number of iterations required by the previous approaches to solve the Kepler time equation, are probably representative of newer methods (e.g., ref. 74) as well as ref. 178.

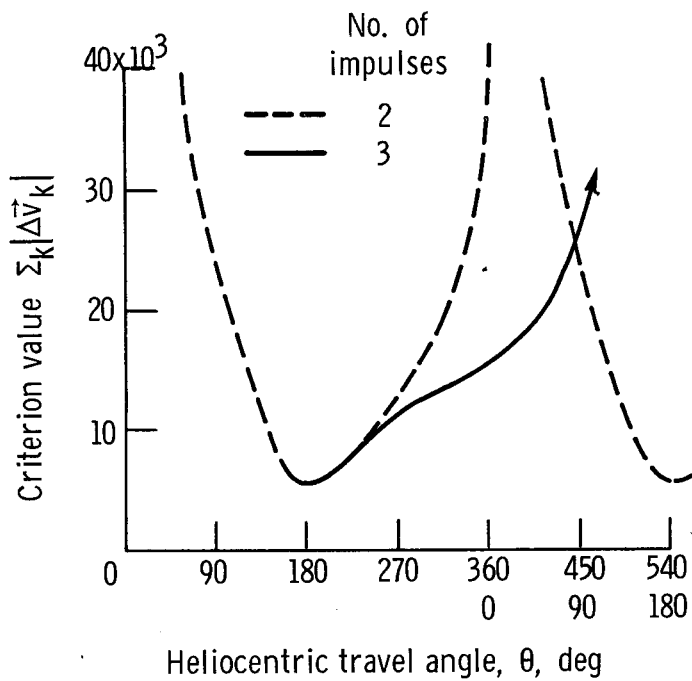
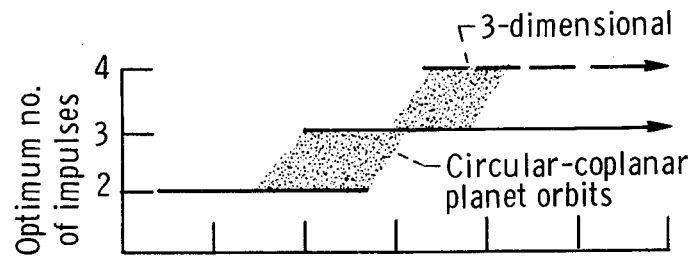


FIGURE 4-15. - COMPARISON OF TWO- AND MULTI-IMPULSE TRAJECTORIES. 250-DAY, EARTH-MARS ORBIT RENDEZVOUS. (CIRCULAR, COPLANAR PLANET ORBITS.)

The present approach has another, less obvious advantage compared even to the most recent available method (ref. 74) in that it appears to be more capable, if not infallible, in determining the optimum number of impulses. This is due primarily to differences in the way that "initial guess" trajectories are prepared. In the present approach the analyst is required to sketch or visualize a plausible-looking trajectory and select the location and timing of 6 (or more) impulses. The descent process then eventually rejects unneeded impulses by driving them to zero magnitude. In the examples shown on Figure 4-15, this led to 2 and 3 impulse trajectories, but (for coplanar planet orbits) never 4, 5, or 6.

Other approaches, by contrast, typically derive an initial guess from a non-optimum (but readily calculable) two-impulse solution. The "primer test" as developed in reference 65 from the basic theory of Lawden (ref. 107) is then applied to determine whether the final trajectory could be improved\* by adding an intermediate impulse. If so, a third impulse is added and its location optimized as in the present approach, e.g., by the Fletcher-Powell technique. (The required fundamental matrices may be computed in closed form, c.f., refs. 33 and 164.) Typically, this results in a locally

---

\* Briefly, Lawden has shown in Chapter 5 of reference 107 that the primer vector ( $\vec{p}$  in the notation of this section) is aligned with the velocity and has unit magnitude at the location of an optimal impulse. If  $|\vec{p}| > 1$  at some point, this implies that the trajectory could be improved by adding an impulse at some interior point, e.g., the point where  $|\vec{p}|$  maximizes.

optimal 3-impulse solution - i.e., whose two subarcs are each locally optimal as judged by the primer test. There are cases, however, in which 4 or more impulses are known to be optimal (c.f., ref. 39); within the context of Earth to Mars orbit rendezvous trajectories this may occur when there is a significant out-of-plane motion.

The cases shown in Figure 4-15 were re-computed for  $270^{\circ} \leq \theta \leq 360^{\circ}$ , under the assumption that Mars is located 0.5 AU above the ecliptic plane. The results, not illustrated, show that: (a) there are in general two distinct locally optimal 3-impulse trajectories, only one of which would be obtained by reference 24 or any other approach relying exclusively as the primer test; and (b) in many cases an (apparently unique) 4-impulse trajectory exists which is somewhat better than either of the 3-impulse solutions. The physical logic for the 4-impulse solution is that energy can be changed most efficiently in a region of high path velocity - i.e., near perihelion as in reference 178 - whereas the plane-change maneuver is accomplished more efficiently in a low velocity region - i.e., farther from the sun as in reference 47. For this reason, earlier results involving 3-impulse, broken-plane transfers (c.f., ref. 35) may be regarded as suspect.

Finally, the MG algorithm was applied to a survey of fast 3-impulse Earth-Neptune trajectories. The results, shown in Figure 4-16, are presented in terms of  $\sum_{k=0}^3 |\Delta \vec{v}|_k$  versus travel time and

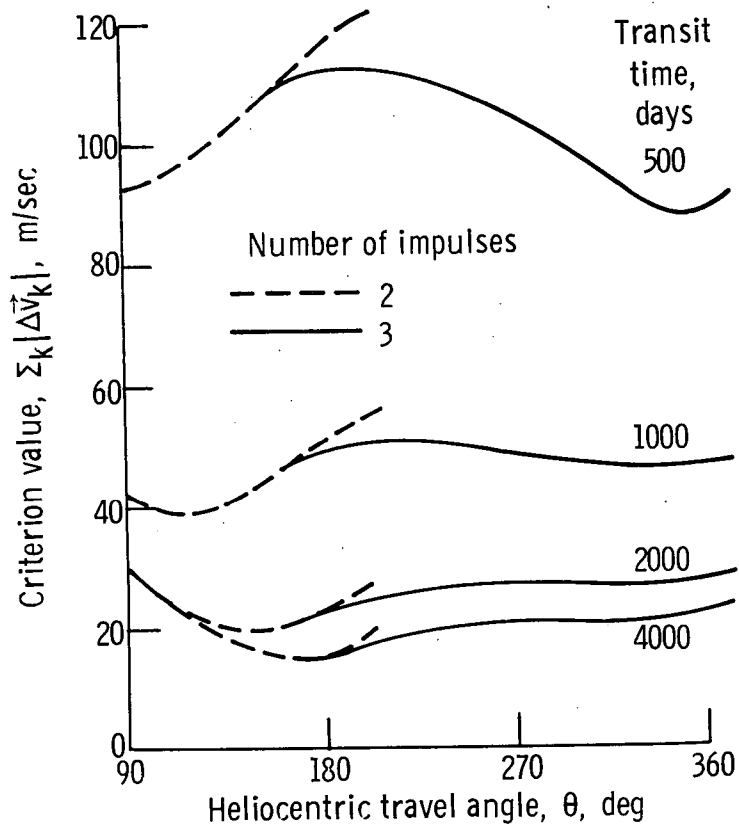


FIGURE 4-16. - CHARACTERISTICS OF OPTIMAL 3-IMPULSE EARTH-NEPTUNE ORBIT RENDEZVOUS TRAJECTORIES. (CIRCULAR, COPLANAR PLANET ORBITS.)

angle as before and required under 2 minutes of IBM 7094 computer time. One entirely unexpected result from this survey are the "second minima" visible in the upper right (at long angles and short times). These correspond to trajectories which nearly graze the sun when the interior impulse is applied. Thus, the velocity is extremely high at the impulse point and a very large energy increment can be gained from a moderate  $|\Delta \vec{v}|_2$ . The point of this example is that: (a) the MG approach is efficient enough to allow the wholesale computational study of optimal multi-impulse trajectories, on a scale previously approached only for 2-impulse non-optimal trajectories; and (b) unexpected, hitherto unsuspected results and conclusions may sometimes emerge from such a study.

## 5. CONCLUDING REMARKS

Although the above-described analysis and examples are preliminary in several respects, they support several basic observations.

(1) First, the MG approach is computationally feasible for its intended class of problems. This further implies that the theoretical structure, developed up to this point, is well founded.

(2) Secondly, the method has realized its basic objective of handling, in a routine manner, problems which are very unstable, and/or involve numerous intermediate boundary conditions. Note that it is not necessary to guess initial values of the adjoint variables.

(3) Also, pointwise state-variable inequality constraints can be incorporated in a routine manner with no sacrifice in run time.

(4) The method has very strong convergence properties in general, and the initial rate of convergence is especially remarkable (recall Table 4-4). Ultimate convergence is quadratic, which is at least as good as that provided by any competitive methods.

(5) As to computational efficiency, the MG method is at least competitive with existing ones for open terminal problems and evidently superior if there is an appreciable number of terminal or intermediate boundary conditions.

(6) The method is characterized by a flexible level structure which can be varied to suit the requirements of a given problem. This has a significant bearing on computational efficiency and suggests the desirability of adding a third, "problem-adaptive" level to the MG method.

Considerable additional work is indicated to confirm and extend these conclusions by applying the MG method to realistic examples drawn from practice. It must also be recognized that the method is presently in an early stage of development. Major generalizations and refinements that could be considered include (but are not limited to) the following:

(1) It would be valuable to have a more general procedure for generating an initial feasible trajectory when control-variable inequality constraints are present. That is, aside from resorting to penalty functions, how do we correct an unfeasible initial trajectory? The notion of generalized iteration parameters (c.f., Section 3.1.3) could usefully be further developed - perhaps by extending the work of reference 115.

(2) Further work in the area of state-variable inequality constraints is required to either develop analytical criteria for satisfactory pointwise approximation to a regional constraint, and/or to incorporate the alternative necessary conditions into the existing Level I and II structure. The technique of reference 158 for the separate computation of arcs lying on a constraint boundary, in

particular, appears to be compatible with the present methods. The constrained arcs (or families thereof) could be treated as "active manifolds" (see 4(a) below).

(3) Criteria and procedures for implementing the problem-adaptive level III should be developed. For instance, a heuristic method might be devised to sub-divide an arc that is "too difficult" (as measured by the number of TPBVP iterations required) and also to combine adjacent "easy" arcs.

(4) Extensions of the MG approach to more general problems than those illustrated would be desirable. For example:

- (a) Problems with active constraint manifolds (e.g., which can produce a discontinuous change in some of the state variables);
- (b) Branched and/or segmented trajectories;
- (c) Problems involving extremely large dimension, e.g.,  $N = 50$ .

(5) There are also numerous detail refinements which can significantly streamline the computations. For one example, it appears to be unnecessary to recompute the transfer matrix at every step. For another the K-matrix data could be used, after the first feasible Level I trajectory has been constructed, to predict initial  $\vec{x}_k$  values corresponding to a chosen state perturbation  $\delta\vec{x}_k$ . Based on limited experience with the system of Example 4.3, this appears to yield a 2 or 3/1 time savings for all Level I steps after the first.

(6) Based on the encouraging results of Example 4-4, it would be worthwhile to further develop the capabilities of the MG method

for approximation purposes. For example, it often happens that simple feedback algorithms or other approximations can be derived which, though perhaps suboptimal, are quite efficient if applied to a short enough problem (the "velocity to gain" rocket steering algorithm is a case in point, the approach of Example 4-4 is another). One of these, used in place of the present lower level iteration procedures, would greatly simplify the calculations by eliminating the need to integrate and store matrix elements. This may be extremely significant for high dimensional problems because the number of equations to be integrated would then vary as  $2N$  rather than  $4N^2$ . It is tempting to speculate that suitable "steering laws" for interesting problems could be derived under the conditions used herein. These, combined with the present results could open a completely new line of approach to high-dimensional problems.

(7) While the MG approach was designed primarily with computational applications in view, it appears to have at least some theoretical utility which could be exploited. For example, necessary conditions for branched or segmented trajectories appear as nearly-obvious corollaries of the basic MG results, but would require laborious derivations if approached from the usual calculus-of-variations viewpoint.

## 6. BIBLIOGRAPHY

The following bibliography is intended neither to be comprehensive nor to imply priority or relative merit. It simply lists the contributions which in some sense influenced the development of the present material. Since not all citations are discussed specifically in the text, it has seemed appropriate to categorize them as shown below.

<u>Description or Major Topic</u>	<u>References</u>
<u>Adaptive Control</u>	79, 89
<u>Approximate Methods</u>	7, 13, 15, 17, 18, 29, 30, 38, 52, 20, 92, 116, 126, 128, 143, 145, 146, 152, 166
<u>Basic Mathematical References</u>	80, 135, 140, 141, 142, 147, 150, 162, 163, 169, 171
<u>Basic Optimization Theory</u> <u>(Texts)</u>	4, 14, 24, 27, 37, 53, 108, 137, 173, 176
<u>Computation Methods</u>	9, 97, 109, 129, 160
<u>Control Iteration</u>	102, 104, 114, 115, 123, 159
<u>Controllability</u>	56
<u>Dynamic and Static Programing</u>	12, 13, 37, 96 through 101, 122
<u>Feedback Control</u>	25, 116
<u>Hamilton-Jacobi Theory</u>	55, 142
<u>Linear System Theory</u>	20, 183

<u>Multilevel and Decomposition Techniques</u>	28, 43, 95, 100, 101, 103, 113, 119, 120, 182
<u>Optimal Control - Misc. Contributions</u>	6, 10, 16, 21, 22, 31, 51, 57, 62, 66, 69, 75, 78, 84, 85, 86, 90, 110, 112, 118, 121, 125, 133, 134, 136, 148, 170, 172, 174, 175, 184
<u>Search Methods</u>	40, 44, 45, 48, 49, 57, 58, 59, 60, 67, 68, 83, 87, 122, 131, 138, 176
<u>Second Variation Methods</u>	19, 25, 123, 124, 148, 161, 167, 177
<u>Singular &amp; Abnormal Problems</u>	41, 71, 76, 82, 91, 156
<u>Space Trajectories</u>	1, 11, 26, 33, 34, 35, 39, 42, 46, 47, 50, 54, 63, 64, 65, 73, 74, 77, 81, 82, 93, 106, 107, 112, 127, 130, 139, 151, 164, 165, 167, 168, 178, 179, 180, 181
<u>Stability</u>	61, 111
<u>State-Variable Inequality Constraints</u>	23, 32, 36, 72, 88, 104, 105, 137, 157, 158, 185
<u>Survey Papers</u>	3, 54, 96, 132, 149
<u>Two-Point Boundary-Value Problems</u>	2, 5, 8, 14, 41, 94

#### References (Listed Alphabetically)

1. Andrus, J. F., Theory of Higher Order Optimum Impulsive Solutions. NASA Contractor Report 1847, May 1971.
2. Armstrong, E. S., A Combined Newton-Raphson and Gradient Parameter Correction Technique for Solution of Optimal-Control Problems, NASA Report L-5786, 1967.
3. Athans, M., The Static of Optimal Control Theory and Applications for Deterministic Systems, IEEE Transactions on Automatic Control, Vol. AC 11, No. 3, pp. 580-592, July 1966.
4. Athans, M. and Falb, P. L., Optimal Control, McGraw-Hill, 1966.
5. Athans, M., Iterative Solution of Optimal Control Problems Via Newton's Method. Lecture Notes, Massachusetts Institute of Technology, July 1968.

6. Athans, M., Falb, P. L., and LaCoss, R. T., Time-, Fuel-, and Energy-Optimal Control of Nonlinear Norm-Invariant Systems, IEEE Transactions on Automatic Control, pp. 196-202, July 1963.
7. Bailey, F. B., The Application of the Parametric Expansion Method of Control Optimization to the Guidance and Control Problem of a Rigid Booster, IEEE Transactions on Automatic Control, Vol. AC-9, No. 1, pp. 74-81, January 1964.
8. Baird, C. D., Jr., Quasilinearization and the Methods of Finite Difference and Initial Values. Journal of Optimization Theory and Applications, Vol. 6, No. 4, pp. 320-330, 1970.
9. Balakrishnan, A. V. and Neustadt, L. W., Computing Methods in Optimization Problems. Academic Press, 1964.
10. Bass, R. W. and Webber, R. F., Optimal Nonlinear Feedback Control Derived from Quartic and Higher-Order Performance Criteria, IEEE Transactions on Automatic Control, Vol. AC-11, No. 3, pp. 448-454, July 1966.
11. Bean, W. C., Noncoplanar Minimum  $\Delta V$  Two-Impulse and Three Impulse Orbital Transfer From a Re-grassing Oblate Earth Assembly Parking Ellipse Onto a Flyby Trans-Mars Asymptotic Velocity Vector, NASA MSC Internal Note No. 69-FM-325, February 1970.
12. Bellman, R. E., Dynamic Programming, Princeton University Press, 1957.
13. Bellman, R. E. and Dreyfus, S., Applied Dynamic Programming, Princeton University Press, 1962.
14. Bellman, R. E. and Kalaba, R. E., Quasilinearization and Non-linear Boundary-Value Problems. American Elsevier Publishing Co., Inc. 1965.
15. Bellman, R. and Richardson, J. M., Linearization Based Upon Differential Approximation and Galerkin's Method, Memorandum RM4614-Pr., The Rand Corporation, July 1966.
16. Berkovitz, L. D., Variational Methods in Problems of Control and Programming, Journal of Mathematical Analysis and Applications, Vol. 3, No. 1, pp. 145-169, August 1961.

17. Birkhoff, G., DeBoor, C. Swartz, B. and Wendroff, B., Rayleigh-Ritz Approximation by Piecewise Cubic Polynomials, J. SIAM Numerical Analysis, Vol. 3, No. 2, 1966.
18. Boyanovitch, D., On the Application of an Approximation Technique To the Problem of Optimal Stabilization of Nonlinear Systems, Research Note 165, Grunman Aircraft Engineering Corp., August 1963.
19. Breakwell, J. V., Speyer, J. L. and Bryson, A. E., Optimization and Control of Nonlinear Systems Using the Second Variation. JSIAM Control, Ser. A, Vol. 1, No. 2, pp. 193-223, 1963.
20. Brockett, R. W., Finite Dimensional Linear Systems, Wiley, 1970.
21. Brown, K. R. and Johnson, G. W., Real-Time Optimal Guidance, IEEE Transactions on Automatic Control, Vol. AC-12, No. 5, pp. 501-506, October 1967.
22. Bryson, A. E. and Denham, W. F., A Steepest-Ascent Method for Solving Optimum Programming Problems, Journal of Applied Mechanics, pp. 247-257, June 1962.
23. Bryson, A. E., Jr., Denham, W. F., and Dreyfus, S. E., Optimal Programming Problems with Inequality Constraints I: Necessary Conditions for Extremal Solutions, AIAA Journal, Vol. 1, No. 11, pp. 2544-2550, November 1963.
24. Bryson, A. E., Jr. and Ho, Y., Applied Optimal Control, Ginn and Company, 1969.
25. Bullock, T. E. and Franklin, G. F., A Second-Order Feedback Method for Optimal Control Computations, IEEE Transactions on Automatic Control, Vol. AC-12, No. 6, pp. 666-673, December 1967.
26. Byrnes, D. V. and Hooper, H. L., Multi-Conic: A Fast and Accurate Method of Computing Space-Flight Trajectories, AIAA Paper No. 70-1062, August 1970.
27. Caratheodory, C., Calculus of Variations and Partial Differential Equations of the First Order. Holden-Day, Inc., 1967.
28. Cavoti, C. R., On the Theory of Optimum Multistage Processes and Its Application", AIAA Paper No. 65-61, January 1965.

29. Chang, C. S. and DeRusso, P. M., An Approximate Method for Solving Optimal Control Problems, IEEE Transactions on Automatic Control, Vol. AC-9, No. 4, pp. 554-556, October 1964.
30. Chang, C. S., Approximate Optimization of Control Systems Using Operations Research. Ph.D. Dissertation, Rensselaer Polytechnic Institute, September 1963.
31. Chuprun, B. E., Solution of Optimization Problems by the Maximum Principle, Automation and Remote Control, No. 9, pp. 1267-1274, September 1967.
32. Chyung, D. M., An Approximation to Bounded Phase Coordinate Control Problem for Linear Discrete Systems, IEEE Transactions on Automatic Control, Vol. AC-12, pp. 37-42, February 1967.
33. Deprit, A., The Matrizants of the Keplerian Motions, Mathematical Note No. 537, Mathematics Research Laboratory, Boeing Scientific Research Lab., October 1967.
34. Dobson, W. F., Huff, V. N., and Zimmerman, A. V., Elements and Parameters of the Osculating Orbit and Their Derivatives, Technical Note D-1106, NASA, January 1962.
35. Doll, J. R., Three-Impulse Trajectories for Mars Stopover Missions in the 1980 Time Period, United Aircraft Research Laboratories, Report F-110398-1, August 1967.
36. Denham, W. F. and Bryson, A. E., Jr., Optimal Programming Problems with Inequality Constraints II: Solution by Steepest-Ascent, AIAA Journal, Vol. 2, No. 1, pp. 25-34, January 1964.
37. Dreyfus, S. E., Dynamic Programming and the Calculus of Variations. Academic Press, 1965.
38. Durbeck, Robert C., An Approximation Technique for Suboptimal Control, IEEE Transactions on Automatic Control, Vol. AC-10, April 1965.
39. Edelbaum, T. N., How Many Impulses?, Astronautics and Aeronautics, pp. 64-69, November 1967.
40. Elkin, R. M., Convergence Theorems for Gauss-Seidel and Other Minimization Algorithms, NASA Technical Report 68-59, January 1968.
41. Falco, M. and Sobierajski, F., Some Remarks Concerning the Newton-Raphson Algorithm and Abnormality in Atmospheric Entry Control Problems, Grumman Research Department Report, March 1966.

42. Fang, B. T. and Brown, J. M., Orbital Guidance and Rendezvous Using Perturbation Methods, AIAA Paper, No. 67-55, January 1967.
43. Fertik, M. A. and Schoeffler, J. D., The Design of Nonlinear Multivariational Control Systems From State Dependent Linear Models. Joint Automatic Control Conference 1965.
44. Fiacco, A. V., Historical Survey of Sequential Unconstrained Methods for Solving Constrained Minimization Problems, N68-18053 Research Analysis Corp., July 1967.
45. Fiacco, A. V. and McCormick, G. P., Nonlinear Programming: Sequential Unconstrained and Minimization Techniques. John Wiley and Sons, Inc., 1968.
46. Fishbach, L. H, Multiple Thrust Level Trajectories for Minimum Propellant Consumption. M.S. Thesis, Case Western Reserve University, November 1967.
47. Fimple, W. R., Optimum Midcourse Plane Changes for Ballistic Interplanetary Trajectories, United Aircraft Corporation Research Laboratories, Report A-110058-3, June 1962.
48. Fletcher, R. and Powell, M. J. D., A Rapidly Convergent Descent Method for Minimization, Computer Journal, Vol. 6, pp. 163-168, July 1963.
49. Fletcher, R. and Reeves, C. M., Function Minimization by Conjugate Gradients, Computer Journal, Vol. 7, pp. 149-154, July 1964.
50. Fraeijs de Veubeke, B. M. and Goorts, J., Optimization of Multiple Impulse Orbital Transfers by the Maximum Principle, Laboratoire d'Aeronautique.
51. Gabasov, R., One Problem in the Theory of Optimal Processes, Automation and Remote Control, No. 8, pp. 1085-1094, August 1967.
52. Gabasov, R. and Kirillova, F. M., Construction of Successive Approximations for Several Optimal Control Problems, Automation and Remote Control, Vol. 27, No. 2, pp. 183-194, February 1966.
53. Gelfand, I. M. and Fomin, S. V., Calculus of Variations, Prentice-Hall, Inc. 1963.
54. Gobetz, F. W. and Doll, J. R., A Survey of Impulsive Trajectories, United Aircraft Research Laboratories, Report G-910557-11, June 1968.

55. Ghaffari, A., On Integration of Hamilton-Jacobi Partial Differential Equation, NASA TM X 55321, August 1965.
56. Gilbert, E. G., Controllability and Observability in Multi-variable Control Systems, J.SIAM Control, Ser. A, Vol. 2, No. 1, 1963.
57. Goldstein, A. A., Convex Programming and Optimal Control, J.SIAM Control, Ser. A, Vol. 3, No. 1, 1965.
58. Goldstein, A. A., On Steepest Descent. J.SIAM Control, Vol 3, No. 1, pp. 147-151, 1965.
59. Guilfoyle, G., Johnson, I., and Wheatley, P., One-Dimensional Search Combining Golden Section and Cubic Fit Techniques, Report No. 67-1, NASA CR-65994, January 1967.
60. Hague, D. S. and Glatt, C. R., An Introduction to Multivariable Search Techniques for Parameter Optimization (And Program Aesop), NASA CR 73200, August 1965.
61. Hahn, W., On the General Concept of Stability and Lyapunov's Direct Method, University of Wisconsin Mathematics Research Center Technical Summary Report No. 485, June 1964.
62. Han, K. W. and Thaler, G. J., Control System Analysis and Design Using a Parameter Space Method, IEEE Transactions on Automatic Control, Vol. AC-11, No. 3, pp. 560-563, July 1966.
63. Hazelrigg, G. A., Jr., Brusch, R. G., and Sachs, W. L., Optimal Space Trajectories with Delta-Velocity Constraints, 21st I.A.F. Congress, Konstanz, West Germany, October 1970.
64. Hazelrigg, G. A., Jr., Optimal Interplanetary Trajectories for Chemically Propelled Spacecraft, AIAA Paper No. 70-1039, August 1970.
65. Hazelrigg, G. A. and Lion, P. M., Analytical Determination of the Adjoint Vector for Optimum Space Trajectories, J. Spacecraft, Vol. 7, No. 10, pp. 1200-1207, 1970.
66. Holtzman, J. M., Convexity and the Maximum Principle for Discrete Systems, IEEE Transactions on Automatic Control, Vol. AC-11, No. 1, pp. 30-35, January 1966.
67. Hestnes, M. R., Multiplier and Gradient Methods, Journal of Optimization Theory and Applications, Vol. 4, No. 5, pp. 303-320, 1969.

68. Hestenes, M. R. and Stiefel, E., Methods of Conjugate Gradients for Solving Linear Systems, Journal of Research of the National Bureau of Standards, Vol. 49, No. 6, pp. 409-436, December 1952.
69. Hull, T. E., A Search for Optimum Methods for the Numerical Integration of Ordinary Differential Equations, SIAM Review, Vol. 9, No. 4, pp. 647-654, 1967.
70. Ivanov, I. N., Stepwise Approximation of Optimum Controls, Prikladnaia Matematika II Mekhanika, Vol. 28, No. 3, pp. 650-656, 1964.
71. Jacobson, D. M., Gershwin, S. B., and Lele, M. M., Computation of Optimal Singular Controls. Technical Report No. 580, Office of Naval Research, January 1969.
72. Jacobson, D. M. and Lele, M. M., A Transformation Technique for Optimal Control Problems with a State Variable Inequality Constraint, IEEE Transactions on Automatic Control, Vol. AC-14, No. 5, pp. 457-465, October 1969.
73. Jezewski, D. J., A Method of Determining Optimal Fixed-Time, N-Impulse Trajectories Between Arbitrarily Inclined Orbits, IAF Paper Ad 30, October 1968.
74. Jezewski, D. J. and Rozendaal, H. L., An Efficient Method for Calculating Optimal Free-Space N-Impulse Trajectories, NASA MSC Internal Note No. 68-FM-135, also TMX 61418, December 1967.
75. Johnson, C. D. and Gibson, J. E., Optimal Control with Quadratic Performance Index and Fixed Terminal Time, IEEE Transactions on Automatic Control, Vol. AC-9, No. 4, pp. 355-369, October 1964.
76. Johnson, C. D. and Gibson, J. E., Singular Solutions in Problems of Optimal Control, IEEE Transactions on Automatic Control, Vol. AC-8, No. 1, pp. 4-15, January 1963.
77. Johnson, F. T., Approximate Finite Thrust Trajectory Optimization.
78. Kalman, R. E., Contributions to the Theory of Optimal Control, Raimpress del Boletin de la Sociedad Matematica Mexicana, 1960.

79. Kalman, R. E., The Variational Principle of Adaptation: Filters for Curve Fitting, Paper presented at IFAC Symposium on Adaptive Control, April 1962.
80. Kantorovich, L. F. and Krylov, V. I., Approximate Methods of Higher Analysis, John Wiley & Sons.
81. Kelley, M. J., Gradient Theory of Optimal Flight Paths, ARS Journal, pp. 947-953, October 1960.
82. Kelley, H. J., Singular Extremals in Lawden's Problem of Optimal Rocket Flight, AIAA Journal, Vol. 1, No. 7, pp. 1578-1582, July 1963.
83. Kelley, Henry J. and Myers, G. E., Conjugate Direction Methods for Parameter Optimization, Paper presented at the 18th Congress of the International Astronautical Federation, September, 1967.
84. Kelley, Henry J., Guidance Theory and Extremal Fields, IRE Transactions on Automatic Control, pp. 75-81, Vol. AC-7, 1962.
85. Kelley, H. J. and Denham, W. F., Examination of Accelerated First Order Methods For Aircraft Flight Path Optimization. NASA Contract Report 66681, October 1968.
86. Kelley, H. J. and Denham, W. F., Modeling and Adjoint for Continuous Systems, Paper presented at the Second International Conference on Computing Methods in Optimization Problems, San Rome, Italy, September 1968.
87. Kelley, H. J., Denham, W. F., Johnson, I. L., and Wheatley, P. O., An Accelerated Gradient Method for Parameter Optimization with Non-Linear Constraints, Journal of the Astronautical Sciences, Vol. 13, pp. 166-169, July-August 1966.
88. Khrustalev, M. M., Sufficient Conditions for Optimality in Problems with Constraints on Phase Coordinates, Automation and Remote Control, No. 4, pp. 544-554, April 1967.
89. Kinnen, E., Study of Adaptive and Non-Linear Control Systems and Theory, Applying the Direct Method of Lyapunov, NASA CR-59689, November 1964.
90. Klee, V., Maximal Separation Theorems for Convex Sets, Memorandum RM-5354-Pr, November 1967.
91. Kopp, R. E. and Moyer, H. G., Necessary Conditions for Singular Extremals, AIAA Paper No. 65-63, January 1965.

92. Krylov, I. A. and Chernous'ke, F. L., On a Method of Successive Approximations for the Solution of Optimal Control Problems, Zhurnal Vychislitel'-Noy Mat. i Mat. Fiz, Vol. 2, No. 6, pp. 1-14, 1962.
93. Kuzmak, G.Ye., Linearized Theory of Optimal Multi-Impulse Coplanar Transfer, Kosmichoskiyo Isslodovaniya (Cosmic Research), Vol. 3, No. 2, pp. 1-37, 1965.
94. Lancaster, E. R., Approximate Solution of a Two-Point Boundary Value Problem, NASA TM X-63139, March 1968.
95. Lefkowitz, I., Multilevel Approach Applied to Control System Design. Joint Automatic Control Conference, 1965.
96. Larson, R. E., A Survey of Dynamic Programming Computational Procedures, IEEE Transactions on Automatic Control, Vol. AC-12, No. 6, pp. 767-774, December 1967.
97. Larson, R. E., Dynamic Programming with Reduced Computational Requirements, IEEE Transactions on Automatic Control, Vol. AC-10, No. 2, pp. 135-143, April 1965.
98. Larson, R. E., State Increment Dynamic Programming, American Elsevier, New York, 1968.
99. Lasdon, L. S., Mathematical Programming. Lecture Notes for Org. 298, Case Western Reserve University, 1965.
100. Lasdon, L. S., Mathematical Programming and Optimal Control, Paper presented to SIAM Journal of Control under the title of "The Conjugate Gradient Method for Optimal Control Problems", 1966.
101. Lasdon, L. S., A Multi-Level Technique for Optimization. Ph.D. Dissertation, Case Western Reserve University, 1964.
102. Lasdon, L. S., Mitter, S. K., and Waren, A. D., The Conjugate Gradient Method for Optimal Control Problems. IEEE Transactions on Automatic Control, Vol. AC-12, No. 2, pp. 132-138, April 1967.
103. Lasdon, L. S. and Schoeffler, J. D., A Multi-Level Technique For Optimization, Joint Automatic Control Conference, 1965.
104. Lasdon, L. S. and Waren, A. D., An Interior Penalty Method for Inequality Constrained Optimal Control Problems. IEEE Transactions on Automatic Control, Vol. AC-12, No. 4, pp. 388-395, August 1967.

105. Lastman, G. J., Optimization of Nonlinear Systems With Inequality Constraints. Ph.D. Dissertation, The University of Texas, January 1967.
106. Lawden, D. F., Minimal Rocket Trajectories, ARS Journal, pp. 360-367, November-December 1953.
107. Lawden, D. F., Optimal Trajectories for Space Navigation, Butterworth Co., Ltd., 1963.
108. Lee, E. B. and Markus, L., Foundations of Optimal Control Theory. John Wiley & Sons, Inc., 1967.
109. Leitmann, G., Optimizatization Techniques. Academic Press, 1962.
110. Levine, M. D., A Steepest Descent Method for Synthesizing Optimal Control Programmes, Paper No. 4, Technical Information Service, AIAA, 1964.
111. Lindorff, D. P. and Monopoli, R. V., Control of Time Variable Nonlinear Multivariable Systems Using Liapunov's Direct Method, NASA Contractor Report 407, March 1966.
112. MacKay, J. S. and Nime, E., Low Variable Thrust Interplanetary Trajectory Data. NASA TN D-4431, April 1968.
113. Macko, D., General Systems Theory Approach to Multi-Level Systems. Ph.D. Dissertation, Case Western Reserve University, 1967.
114. Mehra, R. K. and Bryson, A. E., Conjugate Methods with an Application to V/STOL Flight-Path Optimization, J. Aircraft, Vol. 6, No. 2, pp. 123-128, March-April 1969.
115. Mehra, R. K. and Davis, R. E., A Generalized Gradient Method for Optimal Control Problems with Inequality Constraints and Singular Arcs. IEEE Transactions on Automatic Control, Vol. AC-17, No. 1, pp. 69-79, February 1972.
116. Melsa, J. L. and Schultz, D. G., A Closed-Loop, Approximately Time-Optimal Control Method for Linear Systems, IEEE Transactions on Automatic Control, Vol. AC-12, No. 1, pp. 94-97, February 1967.

118. Merriam, C. W., III, An Optimization Theory for Feedback Control System Design, Information and Control 3, pp. 32-59, 1960.
119. Mesarovic, M., Pearson, J., Macko, D., and Takahara, Y., On the Synthesis of Dynamic Multi-Level Systems, Systems Research Center, Case Institute of Technology, 1965.
120. Mesarovic, M. D., Pearson, J. D., and Takahara, Y., A Multi-level Structure for a Class of Linear Dynamic Optimization Problems. Joint Automatic Contr-1 Conference, 1965.
121. Meschler, P. A., Time-Optimal Rendezvous Strategies, IEEE Transactions on Automatic Control, Vol. AC-8, No. 4, pp. 279-283, October 1963.
122. Miele, A., Levy, A. V., and Cragg, E. E., Modifications and Extensions of the Conjugate Gradient-Restoration Algorithm for Mathematical Programming Problems. Journal of Optimization Theory and Applications, Vol. 7, No. 6, pp. 450-472, 1971.
123. Mitter, S. K., Successive Approximation Methods for the Solution of Optimal Control Problems, Automatica, Vol. 3, pp. 135-149, 1966.
124. Moyer, H. G., Optimal Control Problems That Test for Envelope Contacts., Journal of Optimization Theory and Applications, Vol. 6, No. 4, pp. 287-298, 1970.
125. Nahi, N. E., Design of Time Optimal Systems via the Second Method of Lyapunov, IEEE Transactions on Automatic Control, Vol. AC-9, No. 3, pp. 274-276, July 1964.
126. Nahi, N. E. and Wheeler, L. A., Optimum Terminal Control of Continuous Systems via Successive Approximation to the Reachable Set, IEEE Transactions on Automatic Control, Vol. AC-12, No. 5, pp. 515-521, October 1967.
127. Neustadt, L. W., A General Theory of Minimum-Fuel Space Trajectories. J.SIAM Control, Vol. 3, No. 2, pp. 317-356, 1965.
128. Newland, D. E., On the Methods of Galerkin, Ritz and Krylov-Bogoliubov in the Theory of Non-Linear Vibrations, International Journal of Mechanical Sciences, Vol. 7, pp. 159-172, March 1965.

129. Noton, A. R. M. and Markland, C. A., Numerical Computation of Optimal Control, IEEE Transactions on Automatic Control, Vol. AC-12, No. 1, pp. 59-66, February 1967.
130. Novak, D. H., Computationally Convenient Forms for Conic Section Equations Case 103-9, Bellcomm, Inc., B71 01045, January 1971.
131. Okamura, K., A Simplified Steepest-Ascent Method, Journal of Applied Mechanics, Vol. 33, pp. 452-454, June 1966.
132. Paiewonsky, B., Optimal Control, A Review of Theory and Practice", AIAA Journal, Vol. 3, No. 11, pp. 1985-2006, November 1965.
133. Pearson, J. D., Duality, Linear Filtering and Control, Systems Research Center, Case Institute of Technology.
134. Pearson, J. B., Jr. and Sridhar, R., A Discrete Optimal Control Problem, IEEE Transactions on Automatic Control, Vol. AC-11, No. 2, pp. 171-174, April 1966.
135. Pierce, B. O., A Short Table of Integrals., Ginn and Company, 1956.
136. Plant, J. B., An Iterative Procedure for the Computation of Fixed-Time Fuel-Optimal Controls, IEEE Transactions on Automatic Control, Vol. AC-11, No. 4, pp. 652-660, October 1966.
137. Pontryagin, L. S., Boltyanskii, V. G., Gamkrelidze, R. V., and Mishchenko, E. F., The Mathematical Theory of Optimal Processes. Interscience Publishers, 1962.
138. Powell, M. J. D., An Efficient Method for Finding the Minimum of a Function of Several Variables Without Calculating Derivatives, The Computer Journal, Vol. 7, pp. 155-162, 1964.
139. Ragsac, R. V., Application of Combined High- and Low-Thrust Propulsion Systems to Planetary Missions, Astronautica Acta, Vol. 13, pp. 109-118, 1967.
140. Reza, F. M., Functions of a Matrix, Technical Report No. RADC-TR-67-375, November 1967.
141. Rosen, J. S., The Runge-Kutta Equations by Quadrature Methods, NASA Technical Report R-275, November 1967.
142. Rund, Hanno, The Hamilton-Jacobi Theory in the Calculus of Variations. D. Van Nostrand Company Ltd., 1966.

143. Sakawa, Y., Approximate Solution of Optimal Control Problem by Using Linear Programming Technique, International Aerospace Abstracts, pp. 274-283, October 1964.
145. Saridis, G. N. and Rekasius, Z. V., Design of Approximately Optimal Feedback Controllers for Systems with Bounded States, IEEE Transactions Automatic Control, Vol. AC-12, No. 4, August 1967, pp. 373-379.
146. Saridis, G. N. and Rekasius, Z. V., Design of Approximately Optimal Feedback Controllers for Systems with Bounded States, IEEE Transactions on Automatic Control, Vol. AC-12, No. 4, pp. 373-379, August 1967.
147. Schlechter, S., Quasi-Tridiagonal Matrices and Type-Insensitive Difference Equations, Quarterly of Applied Mathematics, pp. 285-295, October 1960.
148. Schley, C. H., Jr. and Lee I., Optimal Control Computation by the Newton-Raphson Method and the Riccati Transformation IEEE Transactions on Automatic Control, Vol. AC-12, No. 2, pp. 139-144, April 1967.
149. Shankes, E. B., Study of Non-Linear Optimization Techniques. NASA Research Report, September 1970.
150. Shapiro, M. S. and Goldstein, M., A Collection of Mathematical Computer Routines, AEC Research and Development Report, NYO-1480-14, February 1965.
151. Singleton, R. L., Jr., Optimal N-Burn Multiorbit Injection, Thesis for Degree of Master of Science, Massachusetts Institute of Technology, September 1971.
152. Slovikoski, R. D., A Series-Perturbation Technique for Extremum Solutions of Two-Point Boundary Value Problems. Ph.D. Dissertation, The University of Texas at Austin, June 1969.

156. Snow, Donald R., Singular Optimal Controls for a Class of Minimum Effort Problems", J. SIAM Control, Ser. a, Vol. 2, No. 2, pp. 203-219, 1965.
157. Speyer, J. L. and Bryson, A. E., Optimal Programming Problems with a Bounded State Space. AIAA Journal, Vol. 6, No. 8, pp. 1488-1491, August 1968.
158. Speyer, J. L., Mehra, R. K., and Bryson, A. E., Jr., The Separate Computation of Arcs for Optimal Flight Paths With State Variable Inequality Constraints. Technical Report No. 526, Office of Naval Research, May 1967.
159. Stancil, R. T., A New Approach to Steepest-Ascent Trajectory Optimization", AIAA Journal, Vol. 2, No. 8, pp. 1365-1370, August 1964.
160. Sutherland, J. W. and Bohn, E. V., A Numerical Trajectory Optimization Method Suitable for a Computer of Limited Memory", IEEE Transactions on Automatic Control, Vol. AC-11, No. 3, pp. 441-447, July 1966.
161. Tapia, R. A., A Generalization of Newton's Method with an Application to the Euler-Lagrange Equation", PhD. Dissertation, University of California, Los Angeles, October 1967.
162. Tewarson, R. P., A Direct Method for Generalized Matrix Inversion", NASA-CR-79865.
163. Tewarson, R. P., Row-Column Permutation of Sparse Matrices", Computer Journal, Vol. 10, pp. 300-305, November 1967.
164. Teren, F. and Cole, G. L., Analytical Calculation of Partial Derivatives Relating Lunar and Planetary Midcourse Correction Requirements to Guidance System Injection Errors", NASA Technical Note D-4518, April 1968.
165. Thibodeau, J. R., III and Bond, V. R., Launch Window Analysis in a New Prospective with Examples of Departures from Earth to Mars", NASA Technical Note, MSC-04985, September 1971.
166. Vainikko, G., Error Estimates of the Bubnov-Galerkin Method in the Eigenvalue Problem", Vychisl. Mate. i Mate. Fiz, Vol. 5, No. 4, pp. 587-607, 1965.
167. Van Dine, C. P., Extension of the Finite-Difference Newton-Raphson Algorithm to the Simultaneous Optimization of Trajectories and Associated Parameters", AIAA Paper, No. 68-115, January 1968.

168. Van Nhan, N., Rendez-vous Approche Multi-Impulsionnel Optimal, De Grande Amplitude Entre Orbites Elliptiques Proches", Breves Information, pp. 171-174, May-June 1971
169. Van Wyk, R., Variable Mesh Multistep Methods for Ordinary Differential Equations", Research Report 67-12 Rocketdyne, September 1967.
170. Vapnyarskiy, I. B., Some Examples of Solution of Optimal Control Problems by the Variation Method in the Space of the States", December 1965.
171. Varga, R. S., Matrix Iterative Analysis. Prentice-Hall, Inc., 1962.
172. Wells, C. H., Minimum Norm Control of Discrete Systems", IEEE International Convention Record, Vol. 15, Part III, Automatic Control, pp. 55-64, 1967.
173. Weinstock, Robert, Calculus of Variations, McGraw-Hill, 1952.
174. Wonham, W. M. and Johnson, C. D., Optimal Band-Bang Control With Quadratic Performance Index", Journal of Basic Engineering, pp. 107-115, March 1964.
175. Westcott, J. M., Optimal Control Studies", Automatica, Vol. 3, pp. 125-133, 1966.
176. Wilde, D. J., Optimum Seeking Methods. Prentice-Hall, Inc., 1964.
177. Williamson, W. E. and Tapley, B. D., Riccati Transformations for Control Optimization Using the Second Variation", IEEE Transactions on Automatic Control, Vol. AC-17, No. 3, pp. 319-327, June 1972.
178. Willis, E. A., Jr., Optimization of Double-Conic Interplanetary Trajectories", NASA Technical Note D3184, January 1966.
179. Willis, E. A., Jr., Two-Burn Escape Maneuvers With An Intermediate Coasting Ellipse. NASA Technical Note D-5011, February 1969.
180. Willis, E. A., Jr. and Padruft, J. A., Round-Trip Trajectories with Stopovers at Both Mars and Venus", NASA Technical Note D-5758, April 1970.
181. Wilson, S. W., A Pseudostate Theory for the Approximation of Three-Body Trajectories", AIAA Paper, No. 70-1061, August 1970.

182. Yoshida, O., "Optimizing Control Based on the Decomposition Principle", Electrical Engineering in Japan, Vol. 87, pp. 29-38, April 1967.
183. Zadeh, L. A., and DeSoer, S.A., Linear System Theory, McGraw Hill, 1963.
184. Zadeh, L. A., "Optimality and Non-Scalar-Valued Performance Criteria", IEEE Transactions on Automatic Control, Vol. AC-8 No. 1, pp. 59-60, January 1963.
185. Zangwill, W. I., Non-Linear Programming Via Penalty Functions., Management Science, Vol. 13, No. 5, pp. 344-358, January 1967.
186. Zola, C. L., "Trajectory Methods in Mission Analysis for Low-Thrust Vehicles", NASA Lewis TP 20-63, January 1964.

## 7. MAIN SYMBOLS

### Matrices

$A, B, C, D$	$N \times N$ blocks of the canonical (state plus adjoint) equations' fundamental matrix
$E, F, G, H$	derived $N \times N$ matrices
$K$	$KN \times KN$ - dimensional Hessian matrix
$H_k, M_k, N_k$	auxiliary matrices used in the "H-Process" for matrix inversion
$K_i, W_i$	auxiliary matrices used in Varga's recursion formula
$\Phi$	$2N \times 2N$ - dimensional fundamental matrix of the canonical (state plus adjoint) equations

### Vectors

$\dot{\vec{f}}$	time derivative of $N$ -dimensional state vector, c.f., Section 2.1
$\vec{g}$	ditto adjoint vector, c.f., Section 3.1.1
$\vec{k}$	auxiliary vector used in recursion formula
$\vec{s}$	search direction in $R_N$ or $R_{KN}$
$\vec{x}$	$N$ -dimensional state vector, c.f., Section 2.1
$\vec{y}$	$N$ -dimensional state perturbation
$\vec{u}$	$M$ -dimensional control vector
$\vec{v}$	velocity vector, c.f., Section 4.4
$\vec{V}$	control synthesis
$\vec{X}_{pp}$	$NK$ -dimensional state vector in the space of mesh points

$\mathbf{Y}$	NK-dimensional state perturbation vector in the space of mesh points
$\vec{T}$	tangent vector to a surface in $R_N$
$\vec{\psi}$	N-dimensional adjoint vector
$\Delta\vec{\psi}$	adjoint discontinuity at a mesh point
$\Psi$	NK-dimensional adjoint discontinuity vector in mesh point space
$\vec{\phi}$	adjoint perturbation vector
$\vec{\phi}_k, \vec{\sigma}_k, \vec{\gamma}_k, \vec{u}_k$	auxiliary vectors
$\vec{\epsilon}$	terminal error vector
$\vec{\beta}$	L-dimensional vector of design parameters
$\vec{q}$	auxiliary vector
$\vec{v}_k$	auxiliary vector

### Scalars

$\alpha$	(dimensionless) step length in unidimensional search
$\beta$	auxiliary function or bound
$\lambda, \lambda_i, \mu_{ij}$	Lagrangian multipliers
$P_j$	auxiliary variable
$L$	number of function evaluations per step
$\rho$	A bound
$m$	A bound
$t$	independent variable (time)
$\mathcal{H}$	Hamiltonian function
$h_{jk}$	functions describing equality constraints
$J$	criterion value to be minimized

### Subscripts

$i, j, k, l, m, n$	general indices
$o$	original or initial
$opt$	optimal
$f$	final
$u$	due to the control $u(t)$ with $t \in \mathcal{T}$
$des$	desired
$act$	actual
$ext$	extremal
$max$	maximum
$min$	minimum
$L, M, N$	dimension of parameter, control, or state vectors, respectively

### Point Sets

$\mathcal{A}'(\vec{x}_0, t_0; t)$	$N + 1$ dimensional set of attainable states c.f., Section 8.1.1
$\mathcal{R}(t; t_f, \vec{x}_f)$	reversed set of attainable states, c.f., Section 8.1.1
$\mathcal{M}_k$	$k^{th}$ constraint manifold
$F_{sa}$	the set of feasible subarc solutions
$\mathcal{I}$	basic time interval upon which the problem is defined, i.e., $(t_0, t_f)$
$\Omega$	set of admissible controls
$\Gamma$	set of forbidden states
$X_{pp}$	set of permissible mesh points
$R_N$	the real $N$ -tuples
$\pi$	tangent plane to $\mathcal{A}'$ in $R_{N+1}$

Other Notations

$(\vec{\phantom{a}})$	vector
$(\hat{\phantom{a}})$	unit vector
$(\dot{\phantom{a}})$	$d(\phantom{a})/dt$
$(\phantom{a})^{(j)}$	$d^j(\phantom{a})/dt^j$
$(\phantom{a})^T$	transpose of a vector or matrix
$\partial$	boundary of a point set
$\delta, \Delta$	perturbation symbols
$\nabla_z, \nabla_{\vec{z}}$	indicates gradient operation with respect to the variable $z$ or the components of the vector $\vec{z}$

## 8. APPENDIX A

### DISCUSSION OF LEVEL I - THE TWO POINT BOUNDARY VALUE PROBLEM

The general problem described in Eqs. 2.1(1) - 2.1(4) may be reduced to a form suitable for numerical solution by defining the adjoint or costate variables, the Hamiltonian function and then using the Maximum Principle to determine the optimal control law in terms of state and costate variables. The resulting,  $2N$ -dimensional state-adjoint system has mixed or "two-point" boundary conditions - half of which apply at the initial time and half at the final. Such systems cannot, in general, be numerically integrated in one pass - and as pointed out above the required iteration processes are often beset by major difficulties. Thus, to support the development of the "Mesh Gradient" approach in its entirety, it is appropriate to review the theoretical and computational basis of the optimal control TPBVP.

#### 8.1 First-Order Necessary Conditions

In this section, several standard definitions and theorems are presented, without proof, in a form appropriate to the present work and in the interest of completeness and uniformity.

### 8.1.1.1 Basic Definitions and Hypotheses

Continuity and differentiability. - The state-variable derivative function  $\vec{f}(\vec{x}, \vec{u}, \vec{\beta}, t)$  are assumed to be continuous, at least twice-differentiable in the components of  $\vec{x}$  and  $\vec{u}$  and at least once differentiable in  $\vec{t}$  and the components of  $\vec{\beta}$ . This implies that the Hamiltonian function, c.f., Section 8.1.2, has the same properties. Also, the component  $f_0(\vec{x}, \vec{u}, \vec{\beta}, t)$  which defines the integral contribution to the criterion value is regarded as being finite and bounded below.

Admissibility and convexity. - The class of admissible controls, denoted by  $\Omega$ , is the collection of all bounded piecewise continuous controls  $\vec{u}(t)$ , with  $t \in \mathcal{T}$ , whose values lie in the convex set  $\Omega \in R_M$ .

Attainable states. - The set of attainable states in  $R_N$  is defined to be the collection of end points of trajectories  $\vec{x}_u$  emanating from the initial point  $\vec{x}_0(t_0)$ , corresponding to all possible admissible controls  $\vec{u}(t) \in \Omega$ . I.e.,

$$\mathcal{A}(\vec{x}_0, t_0; t) = \{\vec{x}_u(t) | \vec{u}(s) \in \Omega, s \in \mathcal{T}\} \quad (1)$$

The corresponding entity in  $R_{N+1}$  is formed by treating the integral criterion value as an additional state coordinate, say  $x_0 = J$ . It is denoted by the prime notation, i.e.,  $\mathcal{A}'$ .

In a similar fashion the reversed attainability set,

$$\mathcal{A}(t; t_k, \vec{x}_k) = \{\vec{x}_u(t) | \vec{x}_k \in \mathcal{A}(\vec{x}(t), t; t_k)\} \quad (2)$$

is defined as the set of all points  $\vec{x}(t)$  from which the point  $\vec{x}_k = \vec{x}(t_k)$  can be reached.

The notation  $\mathcal{A}_{\text{opt}}(\cdot)$  and  $\mathcal{R}_{\text{opt}}(\cdot)$  will denote the attainability sets corresponding to optimal controls.

It should be noted that the sets  $\mathcal{A}(\vec{x}_0, t_0; t)$  and  $\mathcal{R}(t; t_k, \vec{x}_k)$  are defined at one instant only (namely,  $t$ ). It will also be convenient to define "sets of traversal" as the union, over some subset of  $\mathcal{T}$ , of the sets of attainability. I.e., let

$$\left. \begin{aligned} \mathcal{A}^*(\vec{x}_0, t_0; t) &= \bigcup_{s \in [t_0, t]} \mathcal{A}(\vec{x}_0, t_0; s) \\ \mathcal{R}^*(t; t_k, \vec{x}_k) &= \bigcup_{s \in [t, t_k]} \mathcal{R}(s; t_k, \vec{x}_k) \end{aligned} \right\} \quad (3)$$

and

In passing, it may be observed that although  $\mathcal{A}^*$  and  $\mathcal{R}^*$  are not necessarily convex, they have a nesting property, i.e.,

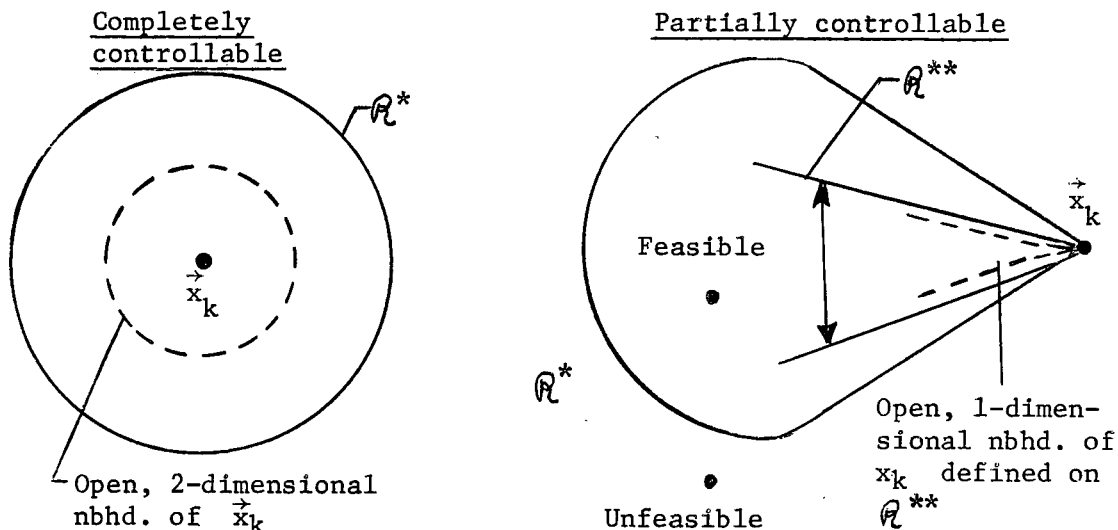
$$\left. \begin{aligned} \mathcal{A}^*(\vec{x}_0, t_0; t_1) &\subset \mathcal{A}^*(\vec{x}_0, t_0; t_2) \\ t_1 &\in [t_0, t_2] \subset \mathcal{T} \end{aligned} \right\} \quad (4)$$

if

with a similar relation holding for the reversed sets.

Controllability. - The system  $\dot{\vec{x}} = \vec{f}(\vec{x}, \vec{u}, \vec{\beta}, t)$  is said to be completely controllable in the neighborhood of a point  $\vec{x}_k \in R_N$  if the traversal set  $\mathcal{R}^*(t; t_k, \vec{x}_k)$  of all points steerable to  $\vec{x}_k(t_k)$  in a finite time interval  $(t_k - t)$  contains an open, N-dimensional neighborhood of  $\vec{x}_k$ . It will be termed partially controllable if  $\mathcal{R}^*$

includes at least one M-dimensional subset  $\mathcal{R}^{**}$  which, in turn, contains an open neighborhood of  $\vec{x}_k$ . These concepts are illustrated by the following sketches.



It is assumed that all systems considered here will be at least partially controllable.

Optimality. - An admissible control  $\vec{u}(t)$  is called optimal if the statement  $\vec{v}(t) \neq \vec{u}(t)$  for some finite subset of  $\mathcal{T}$  implies that  $J(\vec{v}) < J(\vec{u})$ . For practical purposes, optimality is considered to exist when

- (a) The maximum principle and transversality conditions (c.f., Sections 8.1.2 and 8.1.3),
- (b) The convexity or strengthened Legendre-Clebsch condition (c.f., Section 8.2.1),
- (c) The normality condition (Section 8.2.2), and
- (d) The Jacobi or no-conjugate point condition (Section 8.2.3) are satisfied.

The "sufficiency" of these conditions is treated in references 4, 24, and 108, for instance.

### 8.1.2 The Maximum Principle for the Basic Optimal Control Problem

Consider first a simplified version of the problem defined by Eqs. 2.1(1) through 2.1(4), with autonomous dynamics, boundary conditions consisting of two fixed end points and integral criterion only (no PVC). I.e., Problem I:

$$\text{minimize } J = \int_{t_0}^{t_f} f_0(\vec{x}(t), \vec{u}(t)) dt \quad (1)$$

with

$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{u}(t)) \quad (2)$$

subject to

$$\vec{u}(t) \in \Omega \subset R_M \quad (3)$$

with

$$\vec{x}(t_0) = \vec{x}_0 \in R_N \quad (4)$$

and

$$\vec{x}(t_f) = \vec{x}_f \in R_N$$

Let  $f_0, f_1, \dots, f_N$  be continuous functions on the interval  $(t_0, t_f)$ , also possessing continuous first partial derivatives with respect to  $\vec{x}$ . Define the Hamiltonian function by the equation

$$\mathcal{H}(\vec{x}, \vec{\psi}, \vec{u}) = \psi_0 f_0(\vec{x}, \vec{u}) + \vec{\psi} \cdot \vec{f}(\vec{x}, \vec{u}) \quad (5)$$

where  $\vec{x}$  and the adjoint variables  $\vec{\psi}$  are related by the canonical equations

$$\dot{\vec{x}} = \frac{\partial \mathcal{H}}{\partial \vec{\psi}} = \vec{f}(\vec{x}, \vec{u}) \quad (6)$$

$$\dot{\vec{\psi}} = \frac{-\partial \mathcal{H}}{\partial \vec{x}} = -\psi_0 \frac{\partial f_0(\vec{x}, \vec{u})}{\partial \vec{x}} - \left( \frac{\partial \vec{f}(\vec{x}, \vec{u})}{\partial \vec{x}} \right)^T \vec{\psi} \quad (7)$$

Let  $\vec{u}^*(t)$  be a candidate control which is admissible and feasible, i.e., which satisfies Eqs. (3) and (4). Then for  $\vec{u}^*(t)$  to also be optimal, the following conditions must necessarily be satisfied:

(a) There exists a number  $\psi_0 \leq 0$  and an adjoint vector  $\vec{\psi}^*(t)$ , such that  $\vec{\psi}^*(t)$  and  $\vec{x}^*(t)$  are solutions of the canonical equations corresponding to  $\vec{u}^*(t)$ , i.e.,

$$\dot{\vec{x}}^*(t) = \frac{\partial}{\partial \vec{\psi}} \mathcal{H}(\vec{x}^*(t), \vec{\psi}^*(t), \vec{u}^*(t); \psi_0) \quad (8)$$

$$\dot{\vec{\psi}}^*(t) = - \frac{\partial}{\partial \vec{x}} \mathcal{H}(\vec{x}^*(t), \vec{\psi}^*(t), \vec{u}^*(t); \psi_0) \quad (9)$$

satisfying boundary conditions (4) and with a Hamiltonian defined by (5).

(b) The function  $\mathcal{H}(\vec{x}^*(t), \vec{\psi}^*(t), \vec{u}^*(t); \psi_0)$  is maximized with respect to  $\vec{u}(t) \in \Omega$  for  $\vec{u}(t) = \vec{u}^*(t)$ , for all  $t \in \mathcal{T}$ .

That is, given two admissible controls  $\vec{u}^*(t)$  and  $\vec{u}(t)$ , the statements that  $\vec{u}^*(t)$  is optimal while  $\vec{u}(t) \neq \vec{u}^*(t)$  on a finite interval of  $(t_1, t_2)$  imply that

$$\mathcal{H}(\vec{x}^*(t), \vec{\psi}^*(t), \vec{u}(t); \psi_0^*) < \mathcal{H}(\vec{x}^*(t), \vec{\psi}^*(t), \vec{u}^*(t), \psi_0^*) \quad (10)$$

moreover,  $\mathcal{H} = \text{constant} \leq 0$  and the maximum value, 0, occurs when the final time  $t_f$  is free.

### 8.1.3 Extensions to More General Optimal Control Problems

More general cases can easily be imagined, even within the class of two-terminal problems. It will be convenient to classify them as follows:

- (a) System: (1) autonomous; or (2) non-autonomous;
- (b) Criterion: (1) integral; or (2) integral plus PVC;
- (c) Terminal Time: (1) fixed; or (2) free;
- (d) Control: (1) variable,  $\vec{u}(t)$ ; or (2) variable plus fixed parameters,  $\vec{u}(t)$  plus  $\vec{\beta}$ ; and
- (e) Boundary Conditions: (1) fixed or moving target point (N-tuple) in  $R_N$ ,  $\vec{x}_f$  or  $\vec{p}(t_f)$ ; or (2) fixed or moving manifold (k-fold) in  $R_N$ , i.e.,  $g_i(\vec{x}, t) = 0$ ,  $i = 1, 2 \dots N-k$ ; or (3) free end conditions.

Necessary conditions, analogous to and derivable from the results given above for the basic problems, are summarized here for cases of significant interest. For all cases:

- (a) The Hamiltonian is defined as by Eq. (5);
- (b) The state and adjoint variables obey the canonical equations (6) and (7);
- (c) The control satisfies the Maximum principle, Eqs. (9) or (10). It is assumed here that Eqs. (9) or (10) can be used to eliminate explicit appearances of  $\vec{u}(t)$ , so that the canonical equations depend upon  $\vec{x}$ ,  $\vec{\psi}$  and  $t$  only. This may involve, for example, solving the equation

$$\frac{\partial \mathcal{H}}{\partial \vec{u}} = 0 \quad \text{with} \quad \frac{\partial^2 \mathcal{H}}{\partial \vec{u}^2} > 0 \quad (11)$$

constructing "switching boundaries", etc.; and

- (d) The terminal necessary conditions, applicable to the Hamiltonian and the adjoint variables, are summarized in Table 5-1 (pp. 306-307) of reference 24.

It may be observed that the preceding results follow directly from the basic theorem. For example, the non-autonomous case is treated by simply treating time as an additional state variable, i.e.,

$$t = x_{N+1} \quad \text{with} \quad \dot{x}_{N+1} = 1 \quad (12)$$

and then applying the autonomous maximum principle and transversality conditions to the resulting  $N + 1$  dimensional system. Also, note that the terminal conditions shown in Table 5-1 of reference 24 follow directly from the partial-derivative interpretation of the adjoint variables and Hamiltonian; furthermore, these apply (with appropriate changes of sign) to the initial as well as the terminal boundary value.

Necessary conditions for constant design parameters  $\beta_1 \dots \beta_L$  may also be derived, by defining  $L$  additional state variables

$$x_{N+1} \dots x_{N+L} \quad \text{with} \quad \dot{x}_{N+1} = \dot{x}_{N+2} \dots = \dot{x}_{N+L} = 0 \quad (13)$$

and again applying the basic theorem. Note that the Hamiltonian, and hence also the optimal control law, are unaffected by this step. The transversality condition applies to the associated, time-varying adjoint variables  $\psi_{N+1}(t) \dots \psi_{N+L}(t)$  at both the initial and final times. If  $\vec{\beta}$  is unrestricted for example, we have

$$\psi_{N+1}(t_0) = \dots = \psi_{N+L}(t_0) = 0$$

and (14)

$$\psi_{N+1}(t_f) = \dots = \psi_{N+L}(t_f) = 0$$

in addition to all of the previously defined conditions.

## 8.2 Second-Order Necessary Conditions and Sufficient Conditions for a Local Minimum

Under the present assumptions regarding continuity and convexity, it can be shown (c.f., refs. 24 and 108) that local sufficiency can be established by adding the strengthened Legendre-Clebsch, Normality, and Jacobi conditions to the Maximum principle and transversality conditions.

### 8.2.1 Convexity or Legendre-Clebsch Condition

This states that the second-partial matrix

$$\frac{\partial^2 \mathcal{H}}{\partial \vec{u}^2} < 0 \quad (1)$$

(i.e., is negative-definite) for all  $t$  in  $(t_0, t_f)$ . It simply insures that  $\mathcal{H}$  is actually maximized, as in ordinary calculus.

### 8.2.2 Normality Condition

A trajectory leading to the point  $\vec{x}_k$  and lying in the interior of  $\mathcal{R}^*(t; t_k, \vec{x}_k)$  - c.f., Eq. 8.1.1(3) - is called normal. In the Zermelo problem for instance the two envelope lines for  $v < u$  (Fig. 3-3) are in fact the boundary of  $\mathcal{R}^*(t; t_k, \vec{x}_k)$ . A trajectory lying on this boundary clearly does not possess a 2-sided family of neighboring extremals; it is impossible to reach  $\vec{x}_k$  from any

point outside of  $\mathcal{H}^*(t; t_k, x_k)$  in any length of time. Mathematically, this is manifested by the singularity of the matrix  $B$  (c.f., Eqs. 3.2.2(7)). Alternatively, normality is verified if  $B$  is non-singular.

### 8.2.3 The Jacobi or No-Conjugate-Point Condition

The Jacobi condition states that an optimal trajectory may not have a conjugate point at which the constant -  $J$  contours have discontinuous slope, i.e., at which  $\partial^2 J / \partial \vec{x}^2$  is unbounded. Regarding the initial point as fixed it may be seen that the second partial of  $J$ , with respect to variations in the terminal point, is given by Eqs. 3.2.2(8)-(9); i.e.,

$$\left. \frac{\partial^2 J}{\partial \vec{x}^2} \right|_{t_f} = H = DB^{-1} \quad (2)$$

which must be finite. Or with the terminal point fixed, the matrix

$$\left. \frac{\partial^2 J}{\partial \vec{x}^2} \right|_{t_0} = E = -B^{-1}A \quad (3)$$

must be finite.

### 8.2.4 Equivalence of Second-Order Conditions

The above mentioned normality and Jacobi conditions are often seen developed in terms of the backward-sweep approach, c.f., reference 24, rather than in terms of the present forward-sweep or transition-matrix approach. Therefore, it is of some interest to see that the derived matrices  $E$ ,  $F$ ,  $G$ , and  $H$  of Eqs. 4.2.2(9) obey exactly the same Riccatti matrix differential equation as do the

"Gain Matrices" Q, R, U, and S of the backward-sweep approach of reference 24.

The transition-matrix differential Eq. 4.2.2(3) may be written compactly as

$$\frac{d}{dt} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \quad (4)$$

where

$$\alpha = \frac{\partial \dot{\vec{f}}}{\partial \vec{x}}, \quad \beta = \frac{\partial \dot{\vec{f}}}{\partial \vec{\psi}}, \quad \gamma = \frac{\partial \dot{\vec{g}}}{\partial \vec{x}} \quad \text{and} \quad \delta = \frac{\partial \dot{\vec{g}}}{\partial \vec{\psi}} \quad (5)$$

By formally differentiating the defining Eq. 4.2.2(9) for the derived matrices and using the convenient identity that

$$\frac{d}{dt} (B^{-1}) = -B^{-1} \dot{B} B^{-1} \quad (6)$$

it is readily computed that

$$\begin{aligned} E &= F\beta G \\ F &= F\beta H - F\alpha \\ G &= H\beta G - \delta G \\ H &= H\beta H - \delta H - H\alpha - \gamma \end{aligned} \quad (7)$$

These are Riccati differential equations, identical in form to Eqs. 5.3.3.4, 5.3.3.5, 5.3.3.8 and 5.3.3.9 of reference 24.

The latter, in the present  $\alpha$ ,  $\beta$ ,  $\gamma$ , and  $\delta$  notation, may be written:

$$\begin{aligned} \dot{Q} &= u\beta R \\ \dot{U} &= u\beta S - u\alpha \\ \dot{R} &= S\beta R - \delta R \\ \dot{S} &= S\beta S - \delta S - S\alpha - \gamma \end{aligned} \quad (8)$$

Equations (7) and (8) do not possess finite initial conditions and hence, cannot be integrated forward along with the canonical equations. Applicable terminal conditions, however, are given in Chapters 5 and 6 of reference 24. For example, if the terminal boundary condition required is to attain a single fixed point

$$\vec{x}(t_f) = \vec{x}_f$$

then

$$\begin{aligned} Q(t_f) &= 0 \\ U(t_f) &= I \\ R(t_f) &= I \\ S(t_f) &= 0 \end{aligned} \tag{9}$$

Because of the symmetry of Eqs. (8) and 4.2.2(4) and the form of Eq. (7) it is clear that

$$U = R^T \quad \text{and} \quad G = F^T \tag{10}$$

Thus, the present derived matrices would obey the same Ricatti differential equations as those arising in Hamilton-Jacobi theory, but the boundary conditions are different.

The precise relation between the present  $E$ ,  $F$ ,  $G$ , and  $H$  matrices, and  $Q$ ,  $u$ ,  $R$ , and  $S$  of reference 24, may however be displayed by considering the second variation  $\delta^2 J$ . In terms of the present analysis we have that

$$\delta^2 J = [\delta \vec{x}_0, \delta \vec{x}_f] \begin{bmatrix} E & F \\ G & H \end{bmatrix} \begin{bmatrix} \delta x_0 \\ \delta x_f \end{bmatrix} \tag{11}$$

while according to page 183 of reference 24 it is

$$\delta^2 J = [\delta \vec{x}_0, \delta \vec{x}_f] \begin{bmatrix} S - RQ^{-1}R^T & R^T Q^{-1} \\ Q^{-1}R & -Q^{-1} \end{bmatrix} \begin{bmatrix} \delta \vec{x}_0 \\ \delta \vec{x}_1 \end{bmatrix} \quad (12)$$

By comparing (11) and (12) the following relationships evidently hold:

$$E = -B^{-1}A \Big|_{t_0, t} = (S - RQ^{-1}R^T) \Big|_{t, t_0}$$

$$F = B^{-1} \Big|_{t_0, t} = R^T Q^{-1} \Big|_{t, t_0}$$

$$G = (C - DB^{-1}A) \Big|_{t_0, t} = Q^{-1}R \Big|_{t, t_0}$$

$$H = DB^{-1} \Big|_{t_0, t} = -Q^{-1} \Big|_{t, t_0}$$

From this, it is clear that the "normality" and Jacobi conditions as developed in reference 24 using the "backward-sweep" approach, are exactly equivalent to those given here.

### 8.3 Convergence of the Transition Matrix Algorithm

A schematic of the algorithm is shown in Figure 8-1 below.

Its convergence is demonstrated as follows. Let  $\vec{\xi} = \vec{x}(t_f) - \vec{x}_f$ ; this depends on  $\vec{\psi}(t_0)$  only. That is,  $\vec{\xi} = \vec{\xi}(\vec{\psi}_0)$ . To find the roots  $\vec{\psi}_0^*$  of  $\vec{\xi}(\vec{\psi}_0) = 0$ , just expand  $\vec{\xi}$ :

$$\vec{\xi}(\vec{\psi}_0)_{i+1} = \vec{\xi}(\vec{\psi}_0)_i + \delta \vec{\psi}_{0,i} \left[ \frac{\partial \vec{\xi}(\vec{\psi}_0)}{\partial \vec{\psi}} \right]_i \quad (1)$$

or, dropping subscript 0 we have

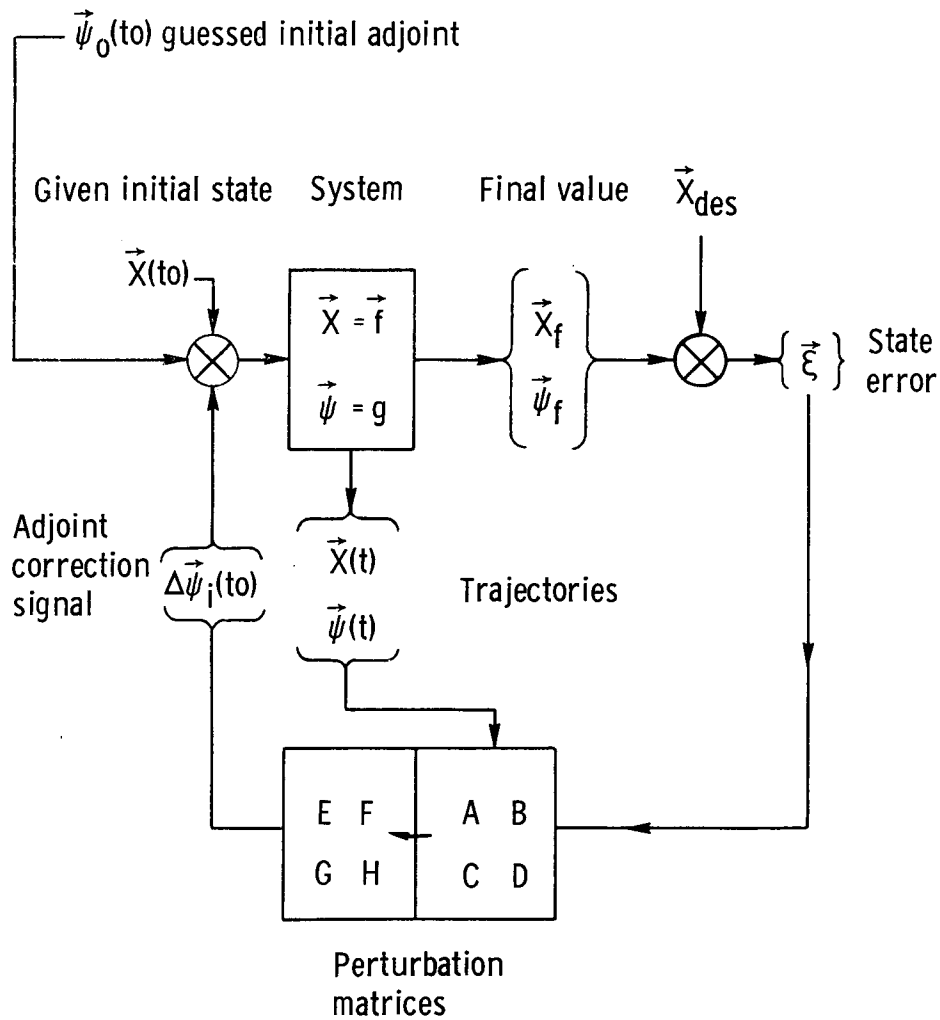


FIGURE 8-1. - SCHEMATIC OF TRANSFER MATRIX ALGORITHM.

$$\vec{\psi}_{i+1} \approx \vec{\psi}_i - \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right]_i^{-1} \vec{\xi}_i \quad (2)$$

Thus

$$\begin{aligned} \vec{\psi}_{i+1} - \vec{\psi}^* &= \vec{\psi}_i - \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right]_i^{-1} \vec{\xi}_i - \vec{\psi}^* = \left( \vec{\psi}_i^T \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right]_i \vec{\xi}_i \right) \\ &- \left( \vec{\psi}^* - \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right]_i \vec{\xi}(\vec{\psi}^*) \right) = \vec{\phi}(\vec{\psi}_i) - \vec{\phi}(\vec{\psi}^*) \end{aligned} \quad (3)$$

Next, expand  $\vec{\phi}$  in Taylor's formula around  $\vec{\psi}^*$ . For the  $j^{\text{th}}$  component we have

$$\vec{\phi}_j(\vec{\psi}_i) = \vec{\phi}_j(\vec{\psi}^*) + (\vec{\psi}_i - \vec{\psi}^*)^T \left[ \frac{\partial \vec{\phi}_j(\vec{\psi}^*)}{\partial \vec{\psi}} \right] + \frac{1}{2} (\vec{\psi}_i - \vec{\psi}^*)^T \left[ \frac{\partial^2 \vec{\phi}_j(\vec{\psi}^*)}{\partial \vec{\psi}^2} \right] (\vec{\psi}_i - \vec{\psi}^*) \quad (4)$$

thus

$$(\vec{\psi}_{i+1} - \vec{\psi}^*)_j = (\vec{\phi}_i - \vec{\phi}^*)_j = (\vec{\psi}_i - \vec{\psi}^*)^T \frac{\partial \vec{\phi}_j}{\partial \vec{\psi}} + \frac{1}{2} (\vec{\psi}_i - \vec{\psi}^*)^T \left[ \frac{\partial^2 \vec{\phi}_j(\vec{\psi}^*)}{\partial \vec{\psi}^2} \right] (\vec{\psi}_i - \vec{\psi}^*) \quad (5)$$

where  $\vec{\psi} = \theta \vec{\psi}_i + (1 - \theta) \vec{\psi}^*$  and  $0 \leq \theta \leq 1$ . But, from (3),

$$\frac{\partial \vec{\phi}}{\partial \vec{\psi}} = [I] - \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}^*} \right]^{-1} \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}^*} \right] + \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}^*} \right]^{-2} \frac{\partial^2 \vec{\xi}}{\partial \vec{\psi}^2} \vec{\xi}(\vec{\psi}^*) = 0 \quad (6)$$

Thus, Eq. (5) leads to the result that for  $i >$  some critical value we have quadratic convergence, i.e.,

$$|\vec{\psi}_{i+1} - \vec{\psi}^*| \leq \rho |\vec{\psi}_i - \vec{\psi}^*|^2 \quad (7)$$

where

$$\rho = \max_j \max_{\theta \in [0,1]} \left\{ \max_{\vec{y} \neq 0} \vec{y}^T \left[ \frac{\partial^2 \phi_1(\vec{\theta}(\theta))}{\partial \vec{\psi}^2} \right] \vec{y} \right\} \quad (8)$$

Directions of descent. - It should be understood that the quadratic convergence displayed above is an ultimate rate, attainable only in some "sufficiently close" neighborhood of the actual root,  $\vec{\psi}^*$ . Thus, initial rates of convergence are of equal concern, because it is evidently possible for a poor choice of  $\vec{\psi}_0$  to cause the Newton algorithm to diverge. Fortunately, the step direction  $\vec{s}$  defined by Eq. (2) is also a direction of descent for the quadratic scalar terminal error function,

$$\frac{1}{2} |\vec{\xi}(\vec{\psi})|^2$$

That is, since

$$\nabla_{\vec{\psi}} \frac{1}{2} |\vec{\xi}(\vec{\psi})|^2 = \vec{\xi}^T \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \quad (9)$$

we have that

$$\nabla_{\vec{\psi}} |\vec{\xi}|^2 \cdot \vec{s} = - \vec{\xi}^T \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right] \left[ \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right]^{-1} \vec{\xi} = - |\vec{\xi}|^2 \leq 0$$

Therefore, by the theorem in Section 3.2, convergence, and hence by implication the quadratic convergence domain can eventually be attained as long as successive directions of descent are well defined. These directions are well defined if the matrices  $\partial \vec{\xi}(\vec{\psi}_n) / \partial \vec{\psi}$  are non-singular. The singular case need not stop the descent if the gradient vector shown in Eq. (9) is non-zero since then an alternative direction of descent is defined. Even at a spurious (non-zero)

local minimum, it may be possible to replace the error function  $|\vec{\xi}|^2$  by a more general form, such as  $\vec{\xi}^T Q \vec{\xi}$ , where  $Q$  is some non-negative matrix, and thereby recover a legitimate direction of descent.

Thus, it may be seen that under fairly general circumstances, the convergence rate of the present algorithm need be no worse than linear, initially, and will ultimately improve to quadratic. Moreover, it is generally possible to have quadratic convergence from the very beginning if the interval of integration  $\Delta t = t_{k+1} - t_k$  is small enough. This may be seen as follows. For the first step, Eq. (7) may be written as

$$|\vec{\psi}_1 - \vec{\psi}^*| \leq \rho |\vec{\psi}_0 - \vec{\psi}^*|^2$$

where the spectral norm,

$$\rho = \max_{\theta \in \mathcal{L} \left\{ \begin{matrix} \vec{\psi}_0 \\ \vec{\psi}^* \end{matrix} \right\}} \left[ \max_{|\vec{y}| \neq 0} \frac{1}{|\vec{y}|} \left[ \vec{y}^T \frac{\partial^2 \phi_1(\vec{\theta}(\theta))}{\partial \vec{\psi}^2} \vec{y} \right] \right] \quad (8a)$$

is the same as the spectral radius of  $\frac{\partial^2 \phi_1}{\partial \vec{x} \partial \vec{\psi}}$  since this matrix is symmetric. It is well known in algebra that the spectral radius of a matrix is bounded by row and column sums, viz

$$\rho \leq \min \left\{ \max_j \sum_{k=1}^n \left| \frac{\partial^2 \phi_1}{\partial \psi_j \partial \psi_k} \right|; \max_k \sum_{j=1}^n \left| \frac{\partial^2 \phi_1}{\partial \psi_j \partial \psi_k} \right| \right\} \quad (10)$$

Referring to the definition of  $\vec{\phi}$  it may be seen that

$$\frac{\partial^2 \phi_1}{\partial \vec{\psi}^2} = -2 \left( \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right)^{-3} \left( \frac{\partial^2 \vec{\xi}}{\partial \vec{\psi}^2} \right) + \left( \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right)^{-2} \left( \frac{\partial^3 \vec{\xi}}{\partial \vec{\psi}^3} \right) + \left( \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \right)^{-2} \left( \frac{\partial^2 \vec{\xi}}{\partial \vec{\psi}^2} \right) \frac{\partial \vec{\xi}}{\partial \vec{\psi}} \quad (11)$$

Because of the definition of  $\vec{\xi}$ , it follows that  $\frac{\partial \vec{\xi}}{\partial \vec{\psi}} = \frac{\partial \vec{x}}{\partial \vec{\psi}}$  and similarly for the higher derivatives. Equations (3a) and (4a) then show that, for short times,

$$\frac{\partial \vec{x}(t)}{\partial \vec{\psi}(t_0)} \approx \frac{\partial \vec{f}}{\partial \vec{\psi}} \cdot \mathbf{I} \cdot \Delta t = \mathcal{O}(\Delta t) \quad (12)$$

Also,

$$\frac{\partial^2 \vec{x}}{\partial \vec{\psi}^2} \approx \frac{\partial^2 \vec{f}}{\partial \vec{\psi} \partial \vec{x}} \frac{\partial \vec{f}}{\partial \vec{\psi}} \Delta t^2 = \mathcal{O}(\Delta t^2)$$

and in a similar fashion

$$\frac{\partial^3 \vec{x}}{\partial \vec{\psi}^3} = \mathcal{O}(\Delta t^3)$$

Thus, each term in Eq. (11) is of order  $\mathcal{O}(\Delta t)$  and, hence vanishes as  $\Delta t \rightarrow 0$ . Therefore,  $\rho$ , which is bounded above by sums of such terms, is also of order  $\mathcal{O}(\Delta t)$ . This means that we can make  $\rho$  as small as we please by choosing a small enough  $\Delta t$ ; the criterion for quadratic convergence can be satisfied in this manner for an arbitrary  $\vec{\psi}_0$  as long as  $\partial \vec{\xi} / \partial \vec{\psi}$  is not singular.

## 9. APPENDIX B

### DISCUSSION OF LEVEL II - SUCCESSIVE IMPROVEMENT STEPS IN STATE SPACE

At this stage, the general problem given in Section 2.1 has been divided into a series of "short" sub-problems by imposing mesh points, and solutions to these short sub-problems have presumably been attained. It has been shown that  $J$  depends on mesh point coordinates only, the gradient of  $J$  in  $X_{pp}$  has been derived, and the "Necessary Conditions" for unconstrained mesh points and for simple constraints have been developed.

#### 9.1 Kuhn-Tucker Necessary Conditions

The celebrated Kuhn-Tucker conditions would apply in the presence of more general constraints on  $X_{pp}$ . See pages 17-34 of reference 45 for a complete discussion. Let the equality-constraint manifolds be described as follows:

$$M_k = \{\vec{x} | h_{jk}(\vec{x}) = 0, j = 1, \dots, n_k\} \quad (1)$$

where

$$k = 0, 1 \dots K + 1 \quad (2)$$

Similarly, the inequality constraint may be described as

$$\Gamma = \{\vec{x} | g_i(\vec{x}) \geq 0, i = 1 \dots m\} \quad (3)$$

Briefly, the first order Kuhn-Tucker condition states that  $J$  is minimized when no feasible perturbation of  $X_{pp}$  will bring about a lower value.

Geometrically, this means that  $\nabla_X J$  must lie within the cone formed by (a) the gradients to the equality-constraint surfaces,  $\nabla_X h_{jk}(X)$ , and (b) the gradients to the binding inequalities,  $\nabla_X g_b(X)$ , (where  $b \in B$ , the set of values of all binding constraints).

Algebraically, for  $\nabla_X J$  to lie within the above described cone, it must be expressible as a linear combination of the gradients to the equality and binding inequality surfaces, i.e.,

$$\nabla_X J = \sum_{b \in B} \lambda_b \nabla_X g_b(X) - \sum_{k=0}^{K+1} \sum_{j=1}^{n_k} \mu_{jk} \nabla_X h_{jk}(X) \quad (4)$$

Here  $\lambda_i$  and  $\mu_{jk}$  are "generalized" Lagrange multipliers, respectively associated with the inequality and equality relations. The inequality multipliers  $\lambda_i$  satisfy the further condition that

$$\lambda_i \geq 0 \quad \text{if} \quad g_i(X) = 0 \quad (\text{constraint binding}) \quad (5)$$

and

$$\lambda_i = 0 \quad \text{if} \quad g_i(X) > 0 \quad (\text{not binding}) \quad (6)$$

Conditions (1) to (6) above, together with a certain regularity assumption (the so-called First Order Constraint Qualification, c.f., p. 19 of ref. 45) comprise the Kuhn-Tucker Necessity Theorem.

## 9.2 Descent Via First-Order Gradient Steps

The "gradient" techniques described (e.g.) in reference 81 have the desirable traits of being conceptually simple, straight forward,

and easy to apply. Unfortunately, initial numerical examples (c.f., Section 4.1) displayed unexpectedly poor initial rates of convergence, especially when a large number of mesh points was used. Investigation showed that this is due to the extreme slopes or curvatures, which tend to develop near a fixed point, having a disproportionate effect on the value of the criterion  $J$ . This, in turn sharply limits the step size that can be taken.

These considerations suggest that descent directions which vanish at the fixed points and which are in some sense as "smooth" as possible, may yield better computational results. Such a direction would allow points far from a fixed terminal to be moved through a relatively large displacement, while those closer to a fixed point are limited to small displacements. In the next section, we derive a direction  $\hat{s}$ , having this quality, as a solution of a variational problem.

### 9.2.1 Auxiliary Variational Problem

Minimize

$$\int_{t_0}^{t_f} \vec{s}^{(n)} \cdot \vec{s}^{(n)} dt \quad \text{where} \quad \vec{s}^{(n)} = \frac{d^n(\vec{s})}{dt^n} \quad (1)$$

subject to the boundary conditions

$$\vec{s}(t_i) = 0, \quad i = 0, 1, 2, \dots, K \quad (2)$$

and the isoperimetric constraint

$$\int_{t_0}^{t_f} \vec{s} \cdot \Delta \vec{\psi} = \text{constant} < 0 \quad (3)$$

Equation (1) guarantees that  $\vec{s}$  will be maximally smooth to order  $n$ ; (2) results in the automatic satisfaction of the fixed point boundary values, and (3) guarantees  $\vec{s}$  to be a direction of descent and provides a common basis for comparing different directions. (Although the problem is formulated for continuous time, with  $\Delta\vec{\psi}(t)$  as defined as in Section 3.2, the formulation and results also apply to the discrete case, i.e., by defining  $\Delta\vec{\psi}$  for Eq. (3) to be an impulse function  $\propto \delta(t - t_k)$ ). Introducing the multiplier  $\lambda$  and writing

$$F(\vec{s}, \vec{s}^{(n)}) = |\vec{s}^{(n)}|^2 + 2\lambda \vec{s} \cdot \Delta\vec{\psi} \quad (4)$$

the Euler-Lagrange equations for this problem become

$$\frac{\partial F}{\partial s_i} + \sum_{j=1}^n (-1)^j \frac{d^j}{dt^j} \left[ \frac{\partial F}{\partial s_i^{(j)}} \right] = 0 \quad (5)$$

and, in case  $n > 1$ , the "natural boundary conditions" (refs. 53 and 173)

$$\left. \frac{d^j}{dt^j} \left( \frac{\partial F}{\partial s^{(n)}} \right) \right|_{t_0} = \left. \frac{d^j}{dt^j} \left( \frac{\partial F}{\partial s^{(n)}} \right) \right|_{t_f} = 0 \quad (6)$$

apply for  $j = 0, 1, \dots, n - 2$ . In terms of the problem at hand, conditions (5) and (6) become

$$\begin{aligned}
 (-1)^{n \rightarrow (2n)} \vec{s} &= -\lambda \Delta \vec{\psi} \\
 \vec{s}(t_0) &= \vec{s}(t_f) = 0 \\
 \vec{s}^{(n)}(t_0) &= \vec{s}^{(n)}(t_f) = 0 \\
 \vec{s}^{(n+1)}(t_0) &= \vec{s}^{(n+1)}(t_f) = 0 \\
 &\dots\dots\dots \\
 \vec{s}^{(2n-2)}(t_0) &= \vec{s}^{(2n-2)}(t_f) = 0
 \end{aligned}
 \tag{7}$$

Thus, we finally obtain  $\vec{s}$  as a polynomial in  $t$ , augmented by a  $2n$ -fold quadrature.

It is of interest to verify that the step  $\vec{s}$  thus defined is indeed a direction of descent. Consider the inner product

$$\begin{aligned}
 p &= \lambda^2 \int_{t_0}^{t_f} \vec{s} \cdot \Delta \vec{\psi} dt \\
 &= \lambda^2 \int_{t_0}^{t_f} \vec{s} \cdot \vec{s}^{(2n)} (-1)^n dt \\
 &= \lambda^2 \left\{ \vec{s} \cdot \vec{s}^{(2n-1)} \Big|_{t_0}^{t_f} - \vec{s}^{(1)} \vec{s}^{(2n-2)} \Big|_{t_0}^{t_f} \right. \\
 &\quad \left. + \vec{s}^{(2)} \vec{s}^{(2n-3)} \Big|_{t_0}^{t_f} - \dots + (-1)^n \int_{t_0}^{t_f} \vec{s}^{(n)} \cdot \vec{s}^{(n)} dt \right\}
 \end{aligned}
 \tag{8}$$

Here again the boundary terms vanish, because of (21) and the integrand is positive definite wherever  $\Delta\vec{\psi} \neq 0$ . Hence, by appropriate choice of the sign of  $\lambda$  we can always achieve that  $p < 0$  (if  $\Delta\vec{\psi} \neq 0$ ), i.e.,  $\hat{s}$  is a direction of descent. Therefore, the theorem in Section 3.2.3 implies the eventual convergence of the following.

### 9.2.2 Linear Step Algorithm

- (a)  $x_0(t)$  = feasible, but otherwise arbitrary initial guess
- (b)  $\Delta\vec{\psi}_j(t)$  = as defined in Section 3.2.2
- (c)  $\vec{s}_j(t)$  = solution of Eqs. (5) to (7) above with  $|\lambda| = 1$
- (d)  $\lambda_j$  = chosen to minimize  $J(\vec{x}_j(t) + \lambda\vec{s}_j(t))$
- (e)  $\vec{x}_{j+1}(t) = \vec{x}_j(t) + \lambda_j\vec{s}_j(t)$
- (f)  $p_j$  = as defined by Eq. (8) above
- (g) Iterate (b) - (f) until the estimated decrement reaches a satisfactorily small magnitude, i. e.,  $0 \leq |p_j| < \epsilon$

### 9.2.3 Ultimate Convergence Rates

Under the conditions applicable in this section, the theorem in Section 3.2.3 implies the ultimate convergence of the preceding algorithm. It remains to examine its rate of ultimate convergence. To do this, note that the recurrence relation  $\vec{s}_{k+1}^{(n)} = -|\lambda|\Delta\vec{\psi}_k$  may be written as

$$\begin{aligned} \vec{s}_{k+1}^{(2n)} = -|\lambda| \left\{ \frac{\partial \vec{h}}{\partial \vec{x}} \cdot (\vec{x}_k + \lambda \vec{s}_k) + \frac{\partial \vec{h}}{\partial \vec{x}(s)} (\vec{x}_k - \lambda \vec{s}_k)^{(j+1)} \right. \\ \left. + \frac{\partial \vec{h}}{\partial t} - g \left[ (\vec{x}_k + \lambda \vec{s}_k), \vec{h}(\vec{x}_k + \lambda \vec{s}_k, \vec{x}_k^j + \lambda \vec{s}_k^j; t) \right] \right\} \quad (1) \end{aligned}$$

That is,  $\vec{s}_{k+1}^{(2n)}$  may be written as a composite function  $\vec{q}$  which depends on  $\vec{s}_k, \dot{\vec{s}}_k, \dots, \vec{s}_k^{(J+1)}$  alone. That is

$$\vec{s}_{k+1}^{(2n)} = \vec{q}(\vec{s}_k, \dot{\vec{s}}_k, \ddot{\vec{s}}_k, \dots, \vec{s}_k^{(J)}, \vec{s}_k^{(J+1)}) \quad (2)$$

This shows that the present algorithm is formally equivalent to Picard iteration. The convergence rate may be exhibited by subtracting the  $k^{\text{th}}$  such equation from the  $k+1^{\text{th}}$ :

$$\vec{s}_{k+1}^{(2n)} - \vec{s}_k^{(2n)} = \vec{q}(\vec{s}_k, \dot{\vec{s}}_k, \ddot{\vec{s}}_k, \dots, \vec{s}_k^{(J)}) - \vec{q}(\vec{s}_{k-1}, \dot{\vec{s}}_{k-1}, \ddot{\vec{s}}_{k-1}, \dots, \vec{s}_{k-1}^{(J)}) \quad (3)$$

If it is now assumed that the composite function  $\vec{q}$  is continuously differentiable with respect to its arguments  $\vec{s}, \dot{\vec{s}}, \dots, \vec{s}^{(J+1)}$ , the mean value theorem for derivatives may be used to show that

$$\begin{aligned} \vec{s}_{k+1}^{(2n)} - \vec{s}_k^{(2n)} &= \frac{\partial \vec{q}}{\partial \vec{s}} \cdot (\vec{s}_k - \vec{s}_{k-1}) + \frac{\partial \vec{q}}{\partial \dot{\vec{s}}} (\dot{\vec{s}}_k - \dot{\vec{s}}_{k-1}) \\ &+ \dots + \frac{\partial \vec{q}}{\partial \vec{s}^{(J+1)}} \cdot (\vec{s}_k^{(J+1)} - \vec{s}_{k-1}^{(J+1)}) \end{aligned} \quad (4)$$

Now define

$$\vec{v}_{k+1} = \vec{s}_{k+1} - \vec{s}_k \quad (5)$$

then

$$\vec{v}_{k+1}^{(2n)} = \frac{\partial \vec{q}}{\partial \vec{s}} \cdot \vec{v}_k + \frac{\partial \vec{q}}{\partial \dot{\vec{s}}} \cdot \dot{\vec{v}}_k + \dots + \frac{\partial \vec{q}}{\partial \vec{s}^{(J+1)}} \vec{v}_k^{(J+1)} \quad (6)$$

Now  $2n$  integrations lead to

$$\vec{v}_{k+1} = \int_0^b \cdots \int_0^b \frac{\partial \tilde{q}}{\partial \vec{s}} \vec{v}_k dt^{2n} + \int_0^b \cdots \int_0^b \frac{\partial \tilde{q}}{\partial \vec{s}} \dot{\vec{v}}_k dt^{2n} + \dots \quad (7)$$

Next, applying the mean value thus for integrals yields

$$\begin{aligned} \vec{v}_{k+1} &= \frac{\partial \tilde{q}}{\partial \vec{s}} \cdot \int_0^b \cdots \int_0^b \vec{v}_k dt^{2n} + \frac{\partial \tilde{q}}{\partial \vec{s}} \int_0^b \cdots \int_0^b \dot{\vec{v}}_k dt^{2n} + \dots \\ &= \frac{\partial \tilde{q}}{\partial \vec{s}} \cdot \int_0^b \cdots \int_0^b \vec{v}_k dt^{2n} + \frac{\partial \tilde{q}}{\partial \vec{s}} \int_0^b \cdots \int_0^b \dot{\vec{v}}_k dt^{2n-1} \\ &\quad + \frac{\partial \tilde{q}}{\partial \vec{s}^{(J+1)}} \int_0^b \cdots \int_0^b \vec{v}_k dt^{2n-J-1} \end{aligned} \quad (8)$$

Finally, by taking magnitudes, using the triangle inequality and properties of the scalar product, and applying the maximum operation yields the result that

$$\begin{aligned} \max |\vec{v}_{k+1}| &\leq \max \left| \frac{\partial \tilde{q}}{\partial \vec{s}} \right| \frac{b^{2n}}{2n!} \max |\vec{v}_k| + \max \left| \frac{\partial \tilde{q}}{\partial \vec{s}} \right| \frac{b^{2n-1}}{(2n-1)!} \max |\dot{\vec{v}}_k| + \dots \\ &\quad \dots + \max \left| \frac{\partial \tilde{q}}{\partial \vec{s}^{(J+1)}} \right| \frac{b^{2n-J-1}}{(2n-J-1)!} \end{aligned} \quad (9)$$

That is,

$$\max_{t \in \mathcal{T}} |\vec{v}_{k+1}| < \max_{t \in \mathcal{T}} |\vec{v}_k| \cdot \sum_{l=0}^{J+1} \max \left| \frac{\partial \tilde{q}}{\partial \vec{s}^{(l)}} \right| \frac{b^{2n-l}}{(2n-l)!} \quad (10)$$

Therefore, assuming that  $2n \geq J + 1$  and that  $\vec{q}$  is sufficiently differentiable, it has been shown that the present algorithm will converge at least geometrically if the interval  $b = t_f - t_0$  is chosen small enough. The result may be extended to longer intervals by partitioning into  $I$  sub-intervals and then extending the maximum operation to include the stipulation that  $i \leq l \leq I$ .

### 9.3 The Hessian Matrix and Descent Via Second-Order Steps

After having selected a set of mesh points  $\vec{x}_k$  and performing the point-point transfers as in Section 3.1, the following information is available at the beginning and end of each arc:

- (a) The values of  $\vec{\psi}(t)$  and  $\Delta\vec{\psi}(t)$
  - (b) The matrices  $A = \frac{\partial \vec{x}(t)}{\partial \vec{x}(t_k)}$ ,  $B = \frac{\partial \vec{x}(t)}{\partial \vec{\psi}(t_k)}$ ,  $C = \frac{\partial \vec{\psi}(t)}{\partial \vec{x}(t_k)}$ , and
- $$D = \frac{\partial \vec{\psi}(t)}{\partial \vec{\psi}(t_k)}$$

- (c) And of course the values of  $\vec{x}(t)$  itself

This same data that was generated in the process of computing the individual two-point transfers can also be used to define a second-order improvement step,  $\vec{y}(t)$ , with the following properties:

- (a) The direction

$$Y = [\vec{y}_1 \dots \vec{y}_K]^T$$

is a direction of descent in  $R_{KN}$ ;

- (b) The sequence  $\vec{x}_j(t) + \vec{y}_j(t)$  converges quadratically to  $\vec{x}_{opt}(t)$  for sufficiently large  $j$  and if

$$\max_t |\vec{x}_0(t) - \vec{x}_{opt}(t)| \text{ is suitably small.}$$

### 9.3.1 The Hessian

As in Section 3.1, the perturbation equations may be written

$$\begin{bmatrix} \vec{y} \\ \vec{\phi} \end{bmatrix}_{k+1} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}_k \begin{bmatrix} \vec{y} \\ \vec{\phi} \end{bmatrix}_k \quad (1)$$

or

$$\begin{bmatrix} \vec{\phi}_k \\ \vec{\phi}_{k+1} \end{bmatrix} = \begin{bmatrix} E & F \\ G & H \end{bmatrix}_k \begin{bmatrix} \vec{y}_k \\ \vec{y}_{k+1} \end{bmatrix} \quad (2)$$

where

$$E_k = -B_k^{-1}A_k, \quad F_k = B_k^{-1} \quad (3)$$

$$G_k = [C - DB^{-1}A]_k \quad \text{and} \quad H_k = D_k B_k^{-1}$$

Applying this at the beginning and end of each subarc we see that,

for the  $\vec{x}_0 - \vec{x}_1$  transfer,

$$\begin{aligned} \vec{\phi}_0^- &= E_0 \vec{y}_0 + F_0 \vec{y}_1 \\ \vec{\phi}_1^+ &= G_0 \vec{y}_0 + H_0 \vec{y}_1 \end{aligned} \quad (4a)$$

Similarly, for the  $\vec{x}_1 - \vec{x}_2$  transfer,

$$\begin{aligned} \vec{\phi}_1^- &= E_1 \vec{y}_1 + F_1 \vec{y}_2 \\ \vec{\phi}_2^+ &= G_1 \vec{y}_1 + H_1 \vec{y}_2 \end{aligned} \quad (4b)$$

after which,

$$\begin{aligned} \vec{\phi}_2^- &= E_2 \vec{y}_2 + F_2 \vec{y}_3 \\ \vec{\phi}_3^+ &= G_2 \vec{y}_2 + H_2 \vec{y}_3 \end{aligned} \quad (4c)$$

and, so forth, until finally, for the  $K$  to  $K+1$  transfer we have



or, more compactly

$$KY = - \vec{\Psi} \quad (8)$$

where  $K$  is the  $KN \times KN$ , block-tridiagonal Hessian matrix indicated above and  $Y$  and  $\Psi$  are as previously defined. Thus, if  $K$  is non-singular the solution

$$Y = - K^{-1} \vec{\Psi} \quad (9)$$

may in principle be computed immediately since the  $\Delta\vec{\Psi}$ 's and the elements of  $K$  are defined in terms of the preceding trajectory.

### 9.3.2 Quadratic Step Algorithm

- (a) The initial mesh,  $\vec{x}_{k,0}$ , is arbitrary, but feasible
- (b) The mesh  $\vec{x}_{k,i}$  is defined at the beginning of iteration number  $i$ . Compute the optimal sub-transfers per Level I.
- (c) Using the  $\Delta\vec{\Psi}$ 's and sub-matrices thus defined, form  $\vec{\Psi}$ ,  $K$ , and then solve for the mesh-point perturbation vector  $Y$
- (d) Convergence is attained if  $|\vec{\Psi}|$  is sufficiently small. If so, stop; otherwise continue
- (e) Define the  $i + 1^{\text{th}}$  mesh as

$$\vec{x}_{k,i+1} = \vec{x}_{k,i} - \vec{y}_{k,i} \quad 1 \leq k \leq K$$

and also estimate initial  $\vec{\Psi}$ 's for the next set of 2-point sub-problems, i.e.,  $\vec{\Psi}(t_{k,i+1}) = \vec{\Psi}(t_{k,i}) + \vec{\phi}_{K,i}$ ,  $0 \leq k \leq K + 1$  by evaluating Eq. (2). Repeat (b) - (e) as necessary.

### 9.3.3 Convergence

Ultimate convergence may be most easily studied in terms of an auxiliary function, which is here taken as  $|\vec{\Psi}|^2$ . The gradient of

this function is

$$\nabla_{\mathbf{X}} |\psi|^2 = \psi^T \left[ \frac{\partial \psi}{\partial \mathbf{X}} \right] = \psi^T \mathbf{K} \quad (1)$$

But the search direction is  $\mathbf{Y} = -\mathbf{K}^{-1}\psi$ , hence

$$\nabla_{\mathbf{X}} |\psi|^2 \cdot \mathbf{Y} = -\psi^T \mathbf{K} \cdot \mathbf{K}^{-1}\psi \leq 0 \quad (2)$$

so that  $\mathbf{Y}$  is indeed a direction of descent for  $|\psi|^2$ . Thus, ultimate convergence (to  $|\psi|^2 \approx 0$ ) is implied by Section 3.2.3 provided that  $\mathbf{K}$  and its component parts are well defined and non-singular.

Indeed, it can be shown that  $|\psi|^2 \rightarrow 0$  quadratically for  $n$  large enough. Consider the recursion formula:

$$\psi_{n+1} = \psi_n - \left( \frac{\partial \mathbf{Y}}{\partial \psi} \right)^{-1}_{\mathbf{n}} \mathbf{Y}_N = \psi_n - \mathbf{K}_n^{-1} \mathbf{Y}_N = \phi(\psi_n) \quad (3)$$

Expand  $\phi(\psi_n)$  in Maclaurin series -

$$\phi(\psi_n) = \psi_n^T \left[ \frac{\partial \phi(0)}{\partial \psi} \right]_{\psi=0} + \frac{1}{2} \psi_n^T \left[ \frac{\partial^2 \phi(\theta)}{\partial \psi^2} \right] \psi_n \quad (4)$$

But

$$\left. \frac{\partial \phi}{\partial \psi} \right|_{\psi=0} = 0 \quad (5)$$

Hence

$$\psi_{n+1} = \psi_n^T \left[ \frac{\partial^2 \phi(\theta)}{\partial \psi^2} \right] \psi_n \quad (6)$$

or, again we have quadratic convergence

$$|\psi_{n+1}| \leq \rho |\psi_n|^2 \quad (7)$$

where

$$\rho = \max_{i, \theta} \left( \max_{|\vec{y}| \neq 0} \frac{1}{|\vec{y}|^2} y^T \left[ \frac{\partial^2 \phi_i(\theta)}{\partial \psi^2} \right] y \right) \quad (8)$$

The convergence of  $|\psi|$  to zero could imply a maximum as well as a minimum, value of  $J$ . For a minimum to occur, the matrix

$$K = \frac{\partial \vec{\psi}}{\partial \vec{X}} \quad (9)$$

must be positive definite, and then the estimated decrement in  $J$  is

$$\Delta J = Y^T \cdot \psi = - Y^T K Y \quad (10)$$

as long as  $|Y| \neq 0$ . In other words, convergence is not only quadratic, but monotonic.

#### 9.4 Descent Via Fletcher-Powell Algorithm

This may be accomplished, for example, with a modified Fletcher-Powell routine. That is, successive directions of descent  $\vec{s}_n$  are defined in  $\vec{X}_{pp}$  by the formula

$$\vec{s}_{n+1} = - \vec{L}_n \delta \vec{\psi}_n \quad (1)$$

where  $\vec{L}_n$  is an appropriately dimensioned positive matrix defined by the recursion

$$\vec{L}_{n+1} = \vec{L}_n + \frac{\vec{\sigma}_n \cdot \vec{\sigma}_n^T}{\vec{\sigma}_n^T \vec{\sigma}_n} - \frac{(\vec{L}_n \vec{\phi}_n)(\vec{L}_n \vec{\phi}_n)^T}{\vec{\phi}_n^T (\vec{L}_n \vec{\phi}_n)} \quad (2)$$

where the change in position is

$$\vec{\sigma}_n = \vec{x}_n - \vec{x}_{n-1}$$

and the change in the gradient is

$$\vec{\phi}_n = \Delta \vec{\psi}_n - \Delta \vec{\psi}_{n-1}$$

The successive step-lengths,  $\alpha_n$ , are predicted from the second-partial data instead of being determined by unidimensional searching. Expanding  $J_{n+1}$  to second order in  $\delta \vec{x}$  yields

$$J_{n+1} = J_n + \Delta \vec{\psi}_n^T \delta \vec{x}_n + \frac{1}{2} \delta \vec{x}_n^T K_n \delta \vec{x}_n \quad (3)$$

Replacing  $\delta \vec{x}_n$  by  $\alpha_n \vec{s}_n$ , equating  $\partial J / \partial \alpha_n$  to zero and solving for  $\alpha_n$  leads to a predicted step length

$$\alpha_n = - \frac{\Delta \vec{\psi}_n^T \vec{s}_n}{\vec{s}_n^T K_n \vec{s}_n} \quad (4)$$

which (to second order) will minimize  $J_{n+1}$  along the  $\vec{s}_n$ -direction. To summarize, the preceding descent algorithm in  $X_{pp}$  consists of the following steps:

- (a)  $L_0$  = arbitrary positive definite matrix, e.g.,  $L_0 = I$
- (b)  $\vec{s}_n = - L_n \Delta \vec{\psi}_n$ , per (1)
- (c)  $\alpha_n$  per (4), or by numerical search if (4) does not lead to a reduction of the criterion  $J$ .
- (d)  $\vec{x}_n = \vec{x}_{n-1} + \alpha_{n-1} \vec{s}_{n-1}$
- (e)  $L_{n+1}$  per (2)

Steps (b) - (e) are repeated until an appropriate measure of convergence has been satisfied.

It can be shown that, for this algorithm,

- (a)  $\lim_{n \rightarrow \infty} L_n = K_n^{-1}$
- (b)  $\lim_{n \rightarrow \infty} \alpha_n = 1$ ; and
- (c) Convergence is quadratic.

See references 48, 49, and 68 for further discussion of this algorithm.

## 10. APPENDIX C

### AN ANALYTICAL APPROXIMATION TO THE FUNDAMENTAL MATRIX FOR SPACE TRAJECTORIES

For the purpose of this section, let the vectors  $\vec{v}_1, \vec{v}_2, \vec{v}_3$ , and  $\vec{v}_4$  denote the unit solutions of Eq. 4.4(10); i.e., they are the column vectors that comprise  $\Phi$ , and their individual component equations for column  $i$  (where  $1 \leq i \leq 4$ ) are

$$\left. \begin{aligned} \dot{\vec{v}}_{1i} &= \vec{v}_{2i} \\ \dot{\vec{v}}_{2i} &= -\omega^2(t)\vec{v}_{1i} + \beta\vec{v}_{4i} \\ \dot{\vec{v}}_{3i} &= \omega^2(t)\vec{v}_{4i} \\ \dot{\vec{v}}_{4i} &= -\vec{v}_{3i} \end{aligned} \right\} \quad (1)$$

with

$$\vec{v}_{ji}(t_k) = \delta_{ji}$$

In second order form:

$$\ddot{\vec{v}}_{4i} = -\omega^2(t)\vec{v}_{4i} \quad (2)$$

$$\ddot{\vec{v}}_{1i} = -\omega^2(t)\vec{v}_{1i} + \beta\vec{v}_{4i} \quad (3)$$

If  $\omega$  were constant, the above equations would yield harmonic solutions. These, however, are quite inaccurate except for trajectories involving only a small variation of radius (and hence,  $\omega$ ).

On the other hand, Eqs. (2) and (3) can be solved analytically for a special, but more general, form of  $\omega(t)$  which can be made to exactly match the "real" function at the beginning and end (or any other two points) of the subarc.

That is, let

$$\frac{\dot{\omega}}{\omega^2} = K, \quad \text{a constant} \quad (4)$$

hence

$$-\frac{1}{\omega} \Big|_{t_k}^{t_{k+1}} = Kt \Big|_{t_k}^{t_{k+1}}$$

or, assuming for convenience that  $t_k = 0$ ,  $\omega(0) = \omega_0$ , and  $\omega(t_{k+1}) = \omega_f$  we find that

$$K = \frac{1}{\omega_0 t_{k+1}} - \frac{1}{\omega_f t_{k+1}}$$

or

$$\omega(t') = \frac{1}{\frac{1}{\omega_0} + \frac{t'}{t_{k+1}} \left( \frac{1}{\omega_f} - \frac{1}{\omega_0} \right)} \quad (5)$$

where  $t' = t - t_k$  is the elapsed time since the beginning of the subarc.

A change of variable. - Now consider the transformation defined by  $t' \rightarrow z$ , where  $\dot{z} = \omega$ . Hence,

$$\frac{d(\quad)}{dt} = \omega \frac{d(\quad)}{dz} \quad (6)$$

and

$$\frac{d^2(\quad)}{dt^2} = \omega^2 \frac{d^2(\quad)}{dz^2} + \omega \frac{d\omega}{dz} \frac{d(\quad)}{dz}$$

Thus, the substitution  $t \rightarrow z$  reduces (2) to the form

$$v_{4i}'' + \omega'/\omega v_{4i}' + v_{4i} = 0 \quad (7)$$

But, since  $\omega'/\omega = \dot{\omega}/\omega^2 = K$  where  $K$  is a constant (c.f., Eqs. (4) and (5)) Eq. (7) reduces to harmonic form

$$v_{4i}'' + K v_{4i}' + v_{4i} = 0 \quad (8)$$

The homogeneous solution. - The solution of (8) is well known to be either

$$v_{4i} = e^{-\frac{Kz}{2}} [A \sin \tau + B_i \cos \tau] \quad (9a)$$

if  $|K| \leq 2$ , or

$$v_{4i} = e^{-\frac{Kz}{2}} [A_i \sinh \tau + B_i \cosh \tau] \quad (9b)$$

if  $|K| > 2$

where

$$\tau = \left[ 1 - \frac{K^2}{4} \right]^{1/2} z = az \quad (10)$$

The oscillatory solution (9a) will be seen to be of primary interest.

Since  $K$  was defined as

$$K = \frac{1}{\omega_0 t_f'} - \frac{1}{\omega_f t_f'}$$

(where  $t_f' = t_{k+1} - t_k$ ) it is clear that  $|K|$  will, in general, be small if  $t_f'$  is large. That is, if

$$t_f > \frac{1}{2} \left| \frac{1}{\omega_0} - \frac{1}{\omega_f} \right| \quad (11)$$

we need never consider (9b). For trips from the Earth inwards,  $\omega_0 = 1$  and  $\omega_f > 1$ , so that (28) is always satisfied easily. That is, we need  $t_f > \frac{1}{2}$  of the Earth's Schuler period (about 29 days) if we are going to 0 A.U.'s ( $\omega_f \rightarrow \infty$ ). For outward transfers, the final angular velocity  $\omega_f$  can become very small, and we must consider that  $t_f$  increases in proportion to the period of the final orbit. This does not seem an unreasonable assumption - however, condition (11) should be verified numerically when "fast" outer-planet trips are being considered. But, for the present we will only consider solution (9a).

The transformed argument. - In order to work with Eq. (9), in either of its forms, we need to know  $z$  as a function of  $t$  and  $\omega$  as a function of  $z$  (as well as  $t$ ). First, let us integrate Eq. (6), i.e.,

$$z(t) = \int_0^t \omega(t) dt \quad (12a)$$

i.e.,

$$z = \frac{t_f}{\left( \frac{1}{\omega_f} - \frac{1}{\omega_0} \right)} \log \left( \frac{\omega_0}{\omega} \right) \quad (12b)$$

or

$$\omega = \omega_0 e^{Kz}$$

The inhomogeneous solution. - By applying the same transformation to Eq. (3) we obtain

$$v_{1i}'' + Kv_{1i}' + v_{1i} = \frac{1}{\omega} v_{4i} \quad (14)$$

where  $v_{4i}$  is the homogeneous solution given in (9) above. First, the complimentary solution of (14) is of the same form as (9) namely

$$v_{1i}^{\text{complimentary}}(z) = \exp\left(\frac{-Kz}{2}\right) [C_i \sin \tau + D_i \cos \tau] \quad (15a)$$

Using the method of variation of parameters we try a particular solution of the form

$$v_{1i}(z) = \exp\left(\frac{-Kz}{2}\right) [U(z) \sin \tau + V(z) \cos \tau] \quad (15b)$$

After some tedious but straight-forward calculations it is found that  $U(z)$  and  $V(z)$  must satisfy the auxiliary differential equations

$$\begin{aligned} V_i' \sin \tau + U_i' \cos \tau &= 0 \\ V_i' \left[ a \cos \tau - \frac{k}{2} \sin \tau \right] - U_i' \left[ a \sin \tau + \frac{k}{2} \cos \tau \right] &= \frac{A_i}{\omega} \sin \tau + \frac{B_i}{\omega} \cos \tau \end{aligned} \quad (16)$$

which leads to the result that

$$\begin{aligned} v_i(z) = \frac{1}{a\omega^2} & \left\{ A_i \left( e^{-2Kz} \left[ \frac{-K \sin 2\tau - a \cos 2\tau}{4(K^2 + a^2)} \right] \right) \right. \\ & \left. + B_i \left( e^{-2Kz} \left[ \frac{-2K \cos \tau + 2a \sin \tau}{4(K^2 + a^2)} \right] \right) - \frac{ae^{-2Kz}}{4(K^2 + a^2)} \right\} \end{aligned} \quad (17)$$

and

$$U(z) = \frac{-1}{a\omega_0^2} \left\{ A_i \left( \frac{e^{-2Kz}}{-4K} - \frac{e^{-2Kz}}{4(K^2 + a^2)} \right) [-K \cos 2Y + a \sin 2\tau] \right. \\ \left. + B_i \left( e^{-Kz} \left[ \frac{K \sin 2\tau - a \cos 2\tau}{4(K^2 - a^2)} \right] \right) \right\} \quad (18)$$

Then, putting (17) and (18) back into (15b), adding the complementary solution (15a) and simplifying we finally obtain

$$v_{1i} = \frac{e^{-\frac{5Kz}{2}}}{4\omega_0^2[K^2 + a^2]} \left\{ A_i \left[ \sin \tau + \frac{a}{K} \cos \tau \right] + B_i \left[ \cos \tau - \frac{a}{K} \sin \tau \right] \right\} \\ e^{-\frac{Kz}{2}} \left\{ C_i \sin \tau + D_i \cos \tau \right\} \quad (19)$$

This together with the homogeneous solution (9) represents the general solution for one column of  $\Phi$ .

The typical column may be written as

$$\vec{v}_i = \begin{bmatrix} v_{1i} \\ \omega v'_{1i} \\ -v'_{4i} \omega \\ v_{4i} \end{bmatrix} \quad (20)$$

with the constants  $A_i \dots D_i$  chosen so that  $\Phi(t_0; t_0) = I$ ; i.e., so that

$$v_{ij}(t_0) = \delta_{ij} \quad (21)$$

Define auxiliary functions  $f_1, f_2, \dots, f_6; g_1, \dots, g_4$  and  $h_1, \dots, h_4$  as follows.

$$\left. \begin{aligned}
 f_1 &= \frac{e^{-\frac{5Kz}{2}}}{4\omega_0^2 [K^2 + a^2]} \\
 f_1' &= \frac{-5Ke^{-\frac{5Kz}{2}}}{8\omega_0^2 [K^2 + a_2]} \\
 f_2 &= \sin \tau + \frac{a}{K} \cos \tau \\
 f_2' &= a \cos \tau - \frac{a^2}{K} \sin \tau \\
 f_3 &= \cos \tau - \frac{a}{K} \sin \tau \\
 f_3' &= -a \sin \tau - \frac{a^2}{K} \cos \tau \\
 f_4 &= e^{-\frac{Kz}{2}} \\
 f_4' &= \frac{-K}{2} e^{-\frac{Kz}{2}} \\
 f_5 &= \sin \tau \\
 f_5' &= a \cos \tau \\
 f_6 &= \cos \tau \\
 f_6' &= -a \sin \tau
 \end{aligned} \right\} \quad (22)$$

$$\left. \begin{aligned}
 g_1 &= f_1 f_2 \\
 g_2 &= f_1 f_3 \\
 g_3 &= f_4 f_5 \\
 g_4 &= f_4 f_6
 \end{aligned} \right\} \quad (23)$$

$$\left. \begin{aligned} h_1 &= g_1' = f_1 f_2' + f_1' f_2 \\ h_2 &= g_2' = f_1 f_3' + f_1' f_3 \\ h_3 &= g_3' = f_4 f_5' + f_4' f_5 \\ h_4 &= g_4' = f_4 f_6' + f_4' f_6 \end{aligned} \right\} \quad (24)$$

Using these functions  $\vec{v}_i$  may be written as

$$\vec{v}_i = \begin{bmatrix} \phi_{1i} \\ \omega \phi_{1i} \\ -\omega \phi_{4i} \\ \phi_{4i} \end{bmatrix} = \begin{bmatrix} g_1^{A_i} & g_2^{B_i} & g_3^{C_i} & g_4^{D_i} \\ \omega h_1^{A_i} & \omega h_3^{B_i} & \omega h_3^{C_i} & \omega h_4^{D_i} \\ -\omega h_3^{A_i} & -\omega h_4^{B_i} & 0 & 0 \\ g_3^{A_i} & g_4^{B_i} & 0 & 0 \end{bmatrix} \quad (25)$$

which may be solved to yield

$$\begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix}_i = \begin{bmatrix} g_1 & g_2 & g_3 & g_4 \\ \omega h_1 & \omega h_2 & \omega h_3 & \omega h_4 \\ -\omega h_3 & -\omega h_4 & 0 & 0 \\ g_3 & g_4 & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} v_{1i} \\ v_{2i} \\ v_{3i} \\ v_{4i} \end{bmatrix} \quad (26)$$

provided that the inverse indicated in Eq. (26) exists.\*

\* That the inverse does exist is readily shown by computing the determinant; i.e.,

$$\begin{aligned} \Delta &= - \begin{vmatrix} g_3 & g_4 \\ h_3 & h_4 \end{vmatrix} \begin{vmatrix} -\omega h_3 & -\omega h_4 \\ g_3 & g_4 \end{vmatrix} = - \left| f_4^2 (f_5 f_6' - f_5' f_6) \right| \\ &= -a^2 e^{-2Kz} = -a^2 \quad \text{when } z = 0 \end{aligned}$$

Since  $a^2 = 1 - K^2/4$  we may rely on the same assumption ( $|K| < 2$ ) by which we selected the oscillatory solution (20a) for further analysis.

The general algebraic form of (25) evaluated at the initial point  $z = 0$ , may be written as

$$\begin{bmatrix} \delta_{1i} \\ \delta_{2i} \\ \delta_{3i} \\ \delta_{4i} \end{bmatrix} = \begin{bmatrix} \frac{a}{4K\omega_0^2[K^2 + a^2]} & \frac{1}{4K\omega_0^2[K^2 + a^2]} & 0 & 1 \\ \frac{-3a\omega}{8\omega_0^2[K^2 + a^2]} & \frac{\omega - \frac{5}{2}K - \frac{a^2}{K}}{4\omega_0^2[K^2 + a^2]} & a\omega & \frac{-K\omega}{2} \\ -a\omega & \frac{K\omega}{2} & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} A_i \\ B_i \\ C_i \\ D_i \end{bmatrix} \quad (27)$$

The solution of (27) is

$$\begin{aligned} B_i &= \delta_{4i} \\ A_i &= \frac{K}{2a} \delta_{4i} - \frac{1}{a\omega} \delta_{3i} \\ D_i &= \delta_{1i} - \frac{\frac{3}{2} \delta_{4i} - \frac{1}{K\omega} \delta_{3i}}{4\omega_0^2[K^2 + a^2]} \end{aligned} \quad (28a)$$

and

$$C_i = \frac{1}{2a\omega} \delta_{2i} + \frac{K}{2a} \delta_{1i} + \frac{1}{4a\omega_0^2[K^2 + a^2]} \left[ 2\delta_{3i} - \left( \frac{3}{4}K - \frac{a^2}{K} + \frac{a}{4} \right) \delta_{4i} \right]$$

It is also of interest to compute the values of  $A_i \dots D_i$  that are appropriate for an impulsive-thrust solution (i.e.,  $\beta = 0$  in Eq.

(3)). Following the procedure shown above, it is readily seen that,

when  $\beta = 0$ , we have

$$\begin{aligned} A_i &= \frac{K}{2a} \delta_{4i} - \frac{1}{a\omega} \delta_{3i} \\ B_i &= \delta_{4i} \\ C_i &= \frac{\delta_{2i}}{2a\omega} + \frac{K}{2a} \delta_{1i} \\ D_i &= \delta_{1i} \end{aligned} \quad (28b)$$

Analytical form of the fundamental matrix. - We have now seen that each of the 4 column vectors  $\vec{v}_i$  ( $i = 1, \dots, 4$ ) composing  $\Phi(z; z_0)$  may be written as

$$\begin{bmatrix} v_{1i} \\ v_{2i} \\ v_{3i} \\ v_{4i} \end{bmatrix} = \begin{bmatrix} g_1 & g_2 & g_3 & g_4 \\ \omega h_1 & \omega h_2 & \omega h_3 & \omega h_4 \\ -\omega h_3 & -\omega h_4 & 0 & 0 \\ g_3 & g_4 & 0 & 0 \end{bmatrix} \begin{bmatrix} A_i \\ B_i \\ C_i \\ D_i \end{bmatrix} \quad (i=1, \dots, 4) \quad (29a)$$

Here the 16 constants  $A_1, \dots, D_4$  are evaluated by applying the Eqs. (22) through (28) successively for  $i = 1, \dots, 4$ . The auxiliary functions  $g_j$  and  $h_j$  ( $j = 1, \dots, 4$ ) are determined by evaluating Eqs. (23) and (24) at the time of interest. That is,

$$\Phi(z; z_0) = \begin{bmatrix} g_1(z) & g_2(z) & g_3(z) & g_4(z) \\ \omega h_1(z) & \omega h_2(z) & \omega h_3(z) & \omega h_4(z) \\ -\omega h_1(z) & -\omega h_4(z) & 0 & 0 \\ g_3(z) & g_4(z) & 0 & 0 \end{bmatrix} \begin{bmatrix} A_1 & A_2 & A_3 & A_4 \\ B_1 & B_2 & B_3 & B_4 \\ C_1 & C_2 & C_3 & C_4 \\ D_1 & D_2 & D_3 & D_4 \end{bmatrix} \quad (29b)$$

Or, renaming the above sets of functions and constants as

$$\tilde{G} = [g_{ik}]$$

and

$$\tilde{C} = [C_{kj}]$$

we may write

$$\Phi(z; z_0) = \tilde{G}\tilde{C}$$

so that

$$\Phi_{ij}(z; z_0) = \sum_k g_{ik} c_{kj}$$

Summary of subarc computations. - (a) Compute the auxiliary function  $\omega(t)$  and  $K$  (Eq. (5));  $z$  and  $\omega(z)$  (Eq. (12));  $a$  and  $\tau$  (Eq. (10)). (b) Evaluate Eqs. (22) through (29) at  $z = 0$  to find the 16 constants  $A_i \dots D_i$  ( $i = 1, \dots, 4$ ) and the 16 components of the fundamental matrix  $\Phi$ . (c) Partition  $\Phi$  onto the A, B, C, and D blocks as in Chapter 3 and, using the given value of  $\beta$  together with the elements of these blocks, solve Eqs. 4.4.1(15) and 4.4.1(13). (d) Equation 4.4.1(11) may be evaluated at intermediate times,  $t_k \leq t \leq t_{k+1}$  to obtain a picture of the state and adjoint trajectory. (e) Numerically compute

$$J = \frac{1}{2} \int_0^{z_f} \left( \psi_4^2 + \psi_5^2 + \psi_0^2 \right) \frac{dz}{\omega(z)}$$

Analytical computation of  $J$ . - As an alternative to numerical integration, it is possible to directly compute that (e.g.)

$$\mu_1(z) = A_1^* g_3(z) + B_1^T g_4(z) \quad (30)$$

where

$$\begin{aligned} A_1^* &= A_1 x_0 + A_2 v_0 + A_3 \lambda_{1,0} + A_4 \mu_{1,0} \\ B_1^* &= B_1 x_0 + B_2 v_0 + B_3 \lambda_{1,0} + B_4 \mu_{1,0} \end{aligned} \quad (31)$$

There are similar expressions for  $\mu_2$  and  $\mu_3$ . Note, however that although the same functions  $g_3(z)$  and  $g_4(z)$  are used for the  $\mu$ 's, the constants (31) are in general different. Hence, let us rewrite (30) and (31) as

$$\mu_j(z) = A_j^* g_3(z) + B_j^* g_4(z)$$

where

$$A_j^* = A_1 x_j(0) + A_2 v_j(0) + A_3 \lambda_j(0) + A_4 \mu_j(0)$$

and

$$B_j^* = B_1 x_j(0) + B_2 v_j(0) + B_3 \lambda_j(0) + B_4 \mu_j(0)$$

Thus,

$$\mu_j^2 = A_j^{*2} g_3^2(z) + 2A_j^* B_j^* g_3(z) g_4(z) + B_j^{*2} g_4^2(z) \quad (32)$$

for  $j = 1 \dots 3$ , so that  $J$  is defined by

$$\begin{aligned} J = & \left( \frac{1}{2} \sum_1^3 A_j^{*2} \right) \int_0^{z_f} g_3^2(z) \frac{dz}{\omega} + \left( \sum_{j=1}^3 A_j^* B_j^* \right) \int_0^{z_f} g_3(z) g_4(z) \frac{dz}{\omega} \\ & + \left( \frac{1}{2} \sum_1^3 B_j^{*2} \right) \int_0^{z_f} g_4^2(z) \frac{dz}{\omega} \end{aligned} \quad (33)$$

The integrals in (33) are

$$\left. \begin{aligned} I_1 &= \int_0^{z_f} \frac{e^{-2Kz} \sin^2(az) dz}{\omega_0} \\ I_2 &= \int_0^{z_f} \frac{e^{-2Kz}}{\omega_0} \sin(az) \cos(az) dz \end{aligned} \right\} \quad (34)$$

and

(Eq. (34) continued on next page)

$$I_3 = \int_0^z \frac{e^{-2Kz}}{\omega_0} \cos^2 ax dz \quad (34)$$

Using the table of integrals, reference 135, these work out to be

$$I_1 = \frac{-e^{-2Kz}}{4K\omega_0} + \frac{e^{-2Kz}}{(4K^2 + 4a^2)\omega_0} (K \cos (2az) - a \sin (2az)) \quad (35)$$

$$I_2 = e^{-2Kz} \left( \frac{-K \sin (2az) - a \cos (2az)}{(4K^2 + 4a^2)\omega_0} \right) \quad (36)$$

$$I_3 = \frac{2e^{-2Kz}}{(4K^2 + 4a^2)\omega_0} \{ \cos az (-aK \cos az + 2a \sin az) \} \quad (37)$$